CONVEX AND STRUCTURED NONCONVEX OPTIMIZATION FOR MODERN MACHINE LEARNING: COMPLEXITY AND ALGORITHMS

A Dissertation
Presented to
The Academic Faculty

By

Digvijay Pravin Boob

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in
Algorithms, Combinatorics, and Optimization

Georgia Institute of Technology

August 2020

CONVEX AND STRUCTURED NONCONVEX OPTIMIZATION FOR MODERN MACHINE LEARNING: COMPLEXITY AND ALGORITHMS

Approved by:

Dr. Guanghui Lan, Advisor
Department of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Santanu S. Dey, Advisor
Department of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Arkadi Nemirovski
Department of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Renato Monteiro
Department of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Santosh Vempala
School of Computer Science
Georgia Institute of Technology

Date Approved: July 15, 2020

To my parents

ACKNOWLEDGMENTS

I am immensely grateful for consistent support of my advisors Guanghui (George) Lan and Santanu Dey throughout my PhD. George was always patient and kind with me, especially in the beginning of my PhD when I was getting introduced to advanced concepts in nonlinear optimization. Santanu has always been mentoring me through different phases of my PhD life and being with him has trained me on how to convey complex ideas in simple words. Discussions with both of them have always boosted my morale to do good quality research. Apart from this, I would also like to thank Rachel Cummings and Praneeth Netrapalli for their support as well as help for my job applications. I am grateful to Renato Monteiro, Arkadi Nemirovski and Santosh Vempala for being part of my thesis committee. I would also like to express my gratitude towards my collaborators: Rachel Cummings, Qi Deng, Yu Gao, Santanu S. Dey, Guanghui Lan, Richard Peng, Saurabh Sawlani, Amaresh Ankit Siva, Uthaipon Tantipongpipat, Charalampos E. Tsourakakis, Chris Waites, Di Wang, Junxing Wang.

I was fortunate to have spent time with good friends at Georgia Tech and I am thankful to Matthew Fahrbach, Sara Kaboudvand, William Kong, Georgios Kotsalis, Arvind Krishna, Kevin Lai, Yulia Lut, Samantha Petti, Adrian Rivera, Samira Samadi, Saurabh Sawlani, Yasaman Shahi, and Peng Zhang. I am also grateful to seniors: Prateek Bhakta, Ben Cousins, Sarah Cannon, Cristóbal Guzmán, Asteroid Santana, Alfredo Torrico, Aurko Roy, and Sadra Yazdanbod. Finally, I would like to thank my parents for their unwavering support and belief in me.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

In this thesis, we investigate various optimization problems motivated by applications in modern-day machine learning. In the first part, we look at the computational complexity of training ReLU neural networks. We consider the following problem: given a fully-connected two hidden layer ReLU neural network with two ReLU nodes in the first layer and one ReLU node in the second layer, does there exists weights of the edges such that neural network fits the given data? We show that the problem is NP-hard to answer. The main contribution is the design of the gadget which allows for reducing the Separation by Two Hyperplane problem into ReLU neural network training problem.

In the second part of the thesis, we look at the design and complexity analysis of algorithms for function constrained optimization problem in both convex and nonconvex settings. These problems are becoming more and more popular in machine learning due to their applications in multi-objective optimization, risk-averse learning among others. For the convex function constrained optimization problem, we propose a novel Constraint Extrapolation (ConEx) method, which uses linear approximations of the constraint functions to define the extrapolation (or acceleration) step. We show that this method is a unified algorithm that achieves the best-known rate of convergence for solving different function constrained convex composite problems, including convex or strongly convex, and smooth or nonsmooth problems with a stochastic objective and/or stochastic constraints. Many of these convergence rates were obtained for the first time in the literature. Besides, ConEx is a single-loop algorithm that does not involve any penalty subproblems. Contrary to existing dual methods, it does not require the projection of Lagrangian multipliers onto a (possibly unknown) bounded set. Moreover, in the stochastic function constraint setting, this is the first

method that requires only bounded variance of the noise; a major relaxation over the restrictive assumption of subgaussian noise in the existing algorithms.

In the third part of this thesis, we investigate a nonconvex nonsmooth function constrained optimization problem, where we introduce a new proximal point method which transforms the initial nonconvex problem into a sequence of convex function constrained subproblems. For this algorithm, we establish the asymptotic convergence as well as the rate of convergence to KKT points under different constraint qualifications. For practical use, we present inexact variants of this algorithm, in which approximate solutions of the subproblems are computed using the aforementioned ConEx method and establish their associated rate of convergence under a strong feasibility constraint qualification.

In the fourth part, we identify an important class of nonconvex function constrained problem for statistical machine learning applications where sparsity is imperative. We consider various nonconvex sparsity-inducing constraints. These are tighter approximations of ℓ_0-norm compared to ℓ_1-norm convex relaxation. For this class of problems, we relax the requirement of strong feasibility constraint qualification to a weaker and a well-known constraint qualification and still prove convergence to KKT points at the rate of gradient descent for nonconvex regularized problems. This work performs a systematic study of the structure of nonconvex sparsity inducing constraints to obtain bounds over Lagrange multipliers and solve certain subproblems faster to achieve convergence rate that matches the rates of nonconvex regularized version under a relaxed constraint qualification which is satisfied by almost all the time.

In the fifth part, we present a faster algorithm for solving mixed packing and covering (MPC) linear programs. The proposed algorithm is from a family of primal-dual type algorithm, similar to ConEx. Here, the main challenge comes from the feasible set of the primal variables being ℓ_∞ ball for a general MPC. The diameter of the ball is at least $\Omega(\sqrt{n})$, where n is the dimension of LP which costs in the complexity. We give specialized treatment to this problem and use a new regularization function which is weaker than strongly convex functions and still obtains accelerated convergence rate. Using this regularizer, we reduce the $\sqrt{n}$ factor in the complexity to $\log n$.

CHAPTER 1
INTRODUCTION

In this chapter, we introduce some background on computational complexity as well as complexity theory for convex optimization which motivated the systematic study of decision and optimization problems.

1.1 Computational Complexity

Computational complexity theory focuses on systematically classifying computational problems into various *complexity classes* based on their inherent difficulty. A computational problem is solved by a computer and is solvable by the application of predefined mathematical steps, i.e., an algorithm. The notion of inherent difficulty is formalized by the amount of resources needed to solve them, such as time and storage, which is known as time complexity and space complexity, respectively. A complexity class is a set of problems with related complexity. The role of computational complexity theory is to determine the practical limits of what computers can and cannot do.

A detailed study of various models of computation is beyond the scope of this chapter. Here, we just state a brief overview of some key complexity classes and the formalism that determines whether a problem belongs to a particular complexity class. In particular, we are interested in four complexity classes for this introduction: P, NP, NP-complete, and NP-hard. Informally, P is a class of problems that can be solved given a deterministic set of rules in $O(\text{poly}(n))$ computations where n is the size of the input, defined appropriately for each problem. The problems in class P are supposed to be efficiently solvable problems. NP is a class of problems that can be solved by the non-deterministic set of rules in time $O(\text{poly}(n))$. It is clear from the description that P is contained in NP. NP-complete is a set of problems that are the hardest in the NP class. This class of problems was introduced by Cook-Levin theorem. To discuss this theorem, the notion of *reduction*

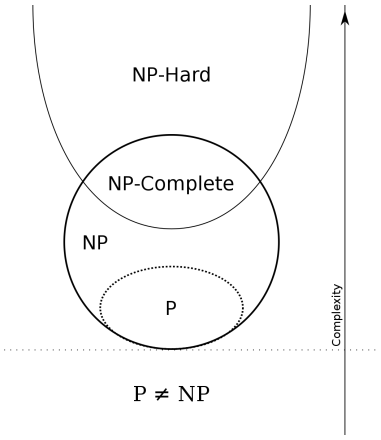


Figure 1.1: Complexity classes P, NP, NP-complete, and NP-hard

becomes important.

Suppose that Problem X and Problem Y are the two classes of problems. We say that *Problem Y can be reduced to Problem X if there is a deterministic polynomial-time method that converts a general instance of Problem Y into a specific instance of Problem X*, which is mathematically denoted as Y $\leqslant_P$ X. Naturally, if there is an algorithm to solve Problem X in polynomial time, then there is an algorithm to solve any general instance of Problem Y. In other words, Y $\leqslant_P$ X implies that Problem X is at least as hard as Problem Y. Cook and Levin independently showed that any problem in NP can be reduced to a set of problems that are called today as NP-complete. Hence, the NP-complete class is the set of hardest problems in the complexity class NP.

NP-hard class is the set of problems that are at least as hard as any problem in NP. Note that they may not be in NP at all. By this description, the simplest way to prove that Problem X is NP-hard is to reduce a general instance of a known NP-complete problem to a particular instance of Problem X in polynomial time. We will use this simple procedure to prove that a certain decision problem related to the training of a neural network is NP-hard. It should be noted that many problems in NP-hard are known to be solved up to some constant approximation ratio or even up to any approximation ratio in polynomial-time. Please refer to Figure 1.1 which summarizes our discussion of the complexity classes.

1.2 Complexity Theory for Convex Optimization

In the previous section, we presented computational complexity theory whose mathematical models of computation are useful for decision problems or sometimes even search problems. However, continuous optimization problems require different models of computation for analyzing their complexity in a meaningful way. In this section, we will look at the complexity theory for general convex optimization problems and understand the limits of what is achievable with commonly used oracles for convex optimization.

A general convex optimization problem can be written as

$$\begin{aligned} f^* := \min_{x \in X} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \leqslant 0, i = 1, \ldots, m, \end{aligned} \tag{1.1}$$

where $X \subset \mathbb{R}^n$ is a convex compact set with nonempty interior, the objective f_0 and constraints $f_i, i = 1, \ldots, m$, are convex continuous functions over X. Let us also assume that convex program (1.1) is feasible and class of such problems is denoted by $\mathcal{C}_m(X)$. Then feasibility assumption along with compactness of X implies optimal value of (1.1) must be attained at some feasible solution, i.e., (1.1) is solvable. We identify an instance of $\mathcal{C}_m(X)$ by $\mathcal{I} = [X, f_0, f_1, \ldots, f_m]$. A first order oracle $\mathcal{G}$ for the class of convex programs, takes an instance $\mathcal{I}$ and a point $x \in \operatorname{int} X$, outputs the values and subgradients of the objective and constraints at the point x. In particular, $\mathcal{G}$ can be defined as a map from X to $\mathbb{R}^{(n+1)\times(m+1)}$ given by

$$x \to \mathcal{G}(x; \mathcal{I}) = [f_0(x), f_0'(x); f_1(x), f_1'(x); \ldots; f_m(x), f_m'(x)].$$

Suppose that a solution mechanism $\mathcal{M}$, applied to instance $\mathcal{I}$, calls the oracle $\mathcal{G}$ sequentially with input x_i, the i-th search point. In the first iteration, the search point x_1 is generated without any information but the i-th search point is generated using accumulated information of the search points already visited. The mechanism can also perform a termination test during the run. How-

ever, the test must depend on the information given by the oracle, $\mathcal{G}$. The final output of mechanism $\mathcal{M}$ on instance $\mathcal{I}$ is denoted by $\bar{x}(\mathcal{I}, \mathcal{M})$. Now that we have introduced sufficient notation, we are ready to talk about complexity for convex optimization. The total number of steps performed by mechanism $\mathcal{M}$, applied to instance $\mathcal{I}$, is called the *iteration complexity*. By iteration complexity, we mean that each iteration involving the evaluation of $\mathcal{G}$ at a certain point, and then doing some simple computation to get the next iterate is considered to be the unit cost. This mode of computation is commonplace for complexity measures involving iterative methods of which mechanism $\mathcal{M}$ is an instance. The use of iterative methods is so mainstream that we denote iteration complexity by just complexity whenever we talk about continuous optimization. We denote the complexity of $\mathcal{M}$ on instance $\mathcal{I}$ by $\text{Compx}(\mathcal{M}, \mathcal{I})$. This quantity can be $+\infty$ if the mechanism does not terminate on instance $\mathcal{I}$. Accordingly, we define the complexity of $\mathcal{M}$ on the family $\mathcal{C}_m(X)$ as

$$\text{Compx}(\mathcal{M}) := \sup_{\mathcal{I} \in \mathcal{C}_m(X)} \text{Compx}(\mathcal{M}, \mathcal{I}).$$

Note that algorithms for convex optimization cannot solve problem (1.1) exactly. However, they can obtain an approximate solution that is reasonably close to the optimal. The closeness to the optimality is denoted by an accuracy measure. Let us denote the accuracy of the solution $x \in X$ for instance $\mathcal{I}$ by,

$$\varepsilon(x; \mathcal{I}) := \max\Big\{ \frac{f_0(x) - f^*}{\max_{y \in X} f_0(y) - f^*}, \frac{\big[f_1(x)\big]_+}{\max_{y \in X} \big[f_1(y)\big]_+}, \ldots, \frac{\big[f_m(x)\big]_+}{\max_{y \in X} \big[f_m(y)\big]_+} \Big\}, \tag{1.2}$$

where $\big[\,\cdot\,\big]_+ := \max\{x, 0\}$. We define the accuracy of $\mathcal{M}$ applied to instance $\mathcal{I}$ by the accuracy of its output $\bar{x}(\mathcal{M}, \mathcal{I})$, i.e.,

$$\text{Accurc}(\mathcal{M}, \mathcal{I}) := \varepsilon(\bar{x}(\mathcal{M}, \mathcal{I}); \mathcal{I}),$$

and the accuracy of mechanism $\mathcal{M}$ applied to the whole family $\mathcal{C}_m(X)$ by

$$\text{Accurc}(\mathcal{M}) := \sup_{\mathcal{I} \in \mathcal{C}_m(X)} \varepsilon(\bar{x}(\mathcal{M}, \mathcal{I}); \mathcal{I}).$$

Finally, the complexity of the family $\mathcal{C}_m(X)$ is defined as the best complexity of a mechanism based on oracle $\mathcal{G}$, for solving problems from this family with a given accuracy, i.e.,

$$\text{Compx}(\varepsilon) = \min_{\mathcal{M}}\{\text{Compx}(\mathcal{M}) : \text{Accurc}(\mathcal{M}) \leqslant \varepsilon\}.$$

Now we look at the lower and upper bounds on the complexity. A lower bound on $\text{Compx}(\varepsilon)$ means for whatever algorithm solving problems in $\mathcal{C}_m(X)$, there always exists a 'bad' problem instance such that number of iterations performed by these algorithms is at least $\text{Compx}(\varepsilon)$. An upper bound on $\text{Compx}(\varepsilon)$ is the number of steps of a particular algorithm that returns a solution of given accuracy for all problems in $\mathcal{C}_m(X)$.

To discuss a major result providing a lower bound for problem class $\mathcal{C}_m(X)$, we introduce one more notion, called *asphericity* κ of X. This term essentially tells how X differs from a Euclidean ball. In particular, the asphericity κ is defined as the smallest ratio of radii of two concentric Euclidean balls V_i and V_o such that $V_i \subseteq X \subseteq V_o$. Below we state the result by Nemirovski and Yudin [79] that provide lower and upper bounds for solving general convex programming problems.

Theorem 1.2.1 *The complexity of the family $\mathcal{C}_m(X)$ of general convex programming problems with m constraints over a convex compact set $X \in \mathbb{R}^n$ of asphericity κ can be bounded by*

$$\min\left\{n, \left\lfloor \frac{1}{(2\kappa\varepsilon)^2} \right\rfloor\right\} \leqslant \textit{Compx}(\varepsilon) \leqslant \left\lceil \frac{4\kappa^2}{\varepsilon^2} \right\rceil, \quad 0 < \varepsilon < 1.$$

We now make the following comments about the above result. First, the upper bound on $\text{Compx}(\varepsilon)$ is obtained by the simple subgradient method. For fixed κ, this upper bound is dimension independent. Second, for high-dimensional problems, i.e., $n \geqslant \left\lfloor \frac{1}{(2\kappa\varepsilon)^2} \right\rfloor$, the lower bound is only a constant factor smaller than the upper bound. Therefore the subgradient method is already optimal for large-scale convex programming problems $\mathcal{C}_m(X)$. The only way to improve the performance of an algorithm is to develop specialized algorithms for important subclasses of $\mathcal{C}_m(X)$. In the next subsection, we will see an optimal method for convex optimization.

1.2.1 Composite convex optimization

In this section, we discuss the convex optimization problem

$$\min_{x \in X} \{\psi(x) := f(x) + \chi(x)\}, \tag{1.3}$$

where we impose the requirement that $f : X \to \mathbb{R}$ has Lipschitz continuous gradients, i.e.,

$$\|\nabla f(x) - \nabla f(y)\|_* \leqslant L\|x - y\|$$

for all $x, y \in X$. Here, $\|\cdot\|_*$ denotes the dual norm and say that f is L-Lipschitz smooth function. We also assume that χ is a convex, possible nonsmooth function satisfying

$$|\chi(x) - \chi(y)| \leqslant M\|x - y\|_*.$$

Hence, (1.3) is a general nonsmooth convex optimization problem. However, there is an additional structure to the problem since we assume that, f, a component of the objective function is *Lipschitz smooth*. Lipschitz smooth means gradients are Lipschitz continuous. Since the objective is composed of two convex components hence, we say that this is a composite convex optimization problem.

Observe that problem (1.3) covers several important classes of convex programming problems as certain special cases. For the sake of simplicity, we assume in the following discussion that the domain X is a standard Euclidean ball.

Non-smooth convex optimization: Suppose that the smooth component $f = 0$ in ψ. Then, problem (1.3) becomes the generic non-smooth convex optimization problem that has been well-studied in the literature. According to Nemirovski and Yudin [79], if the dimension n is sufficiently large, then the complexity of any iterative algorithm satisfied the lower bound

$$\text{Compx}(\varepsilon) \geqslant \frac{M^2}{\varepsilon^2}.$$

Moreover, the simple subgradient descent method can achieve, up to a constant factor, the above lower bound. Nemirovski and Yudin [79] also developed the mirror descent algorithm that can be advantageous over the subgradient descent method when X is not a Euclidean ball by using a prox-function (also called Bregman's distance. More on this will come later).

Smooth convex optimization: Suppose that the non-smooth component $\chi = 0$ in ψ. Then, problem (1.3) becomes the smooth convex optimization problem. In [79], Nemirovski and Yudin show that, if the dimension n is sufficiently large, then the complexity of any iterative algorithm can be lower bounded by

$$\text{Compx}(\varepsilon) \geqslant \left(\frac{L}{\varepsilon}\right)^{1/2}.$$

In a major work, Nesterov [80] showed an upper bound on the complexity which is at most a constant factor worse than the aforementioned lower bound. Hence, it is an optimal method. Nesterov's method was also studied using Bregman distance. However, it is unclear whether the method converges in presence of nonsmooth component χ.

Finally, note that subgradient method when applied to composite convex optimization has the complexity upper bound

$$\text{Compx}(\varepsilon) \leqslant \frac{L^2 + M^2}{\varepsilon^2},$$

whose dependence on Lipschitz constant L is suboptimal. In particular, a trivial lower bound on complexity of (1.3) from [79] is

$$\text{Compx}(\varepsilon) \geqslant \left(\frac{L}{\varepsilon}\right)^{1/2} + \frac{M^2}{\varepsilon^2}.$$

This motivated study of a specialized method that has unified convergence for composite problem (1.3) which we will look in Section 1.3.1.

Note that problem (1.3) is set constrained optimization problem for which most algorithms, e.g., projected gradient descent (PGD), assume that projection is easy to perform. However, most convex sets, represented by constraints $f_1, \ldots, f_m$ in (1.1) are not very simple to justify such an assumption. Indeed, for the simple case of linear constraints, projection operation demands a so-

lution to quadratic programming problem which can be hard to approximate easily. Theorem 1.2.1 provides lower complexity bounds for general nonsmooth convex programming problem (1.1). Below, we provide another interesting result from [86] that proves tighter lower bounds for Lipshitz smooth convex programming problems. In particular, we assume that f_0 is a convex function which is L_{f_0}-Lipschitz smooth, and $f_i, i = 1, \ldots, m$, are linear constraints which contains the case of smooth convex programming problem. Then the problem (1.1) can be written as follows:

$$\begin{aligned} f^* := \min_{x \in X} \quad & f_0(x) \\ \text{subject to} \quad & Ax \leqslant b. \end{aligned} \tag{1.4}$$

Theorem 1.2.2 *The complexity of the family of smooth convex optimization problems* (1.4) *can be lower bounded by*

$$\mathit{Compx}(\varepsilon) \geqslant \max\left\{ \frac{L_{f_0}\|x^*\|^2}{\sqrt{\varepsilon}} + \frac{\|A\| \cdot \|x^*\| \cdot \|y^*\|}{\varepsilon}, \frac{\|A\| \cdot \|x^*\|}{\varepsilon} \right\}.$$

Compared to the lower bounds in Theorem 1.2.1, above theorem provides for lower bounds which are much smaller for the smooth convex optimization problem. It shows that due to the imposed smoothness structure of the class of problems, there is a scope for faster algorithms. For the case when f_0 is *strongly convex*, [86] shows that even smaller bounds of $\Omega(\frac{1}{\sqrt{\varepsilon}})$ can be established on the complexity. Indeed, we will see in Chapter 3 that such lower bound for strongly convex problem can be achieved by a primal-dual type of method. Moreover, unified complexity for composite convex optimization problem of type (1.1) will also be established in Chapter 3.

In the next section, we look at convex optimization with stochastic first-order oracle.

1.3 Convex Optimization under a Stochastic First-order Oracle

In the previous section, we reviewed some important results for convex optimization under exact first-order information. In many situations, the information returned by the first-order oracle is

inexact. One prominent example is given in the following stochastic programming problem:

$$\min_{x \in X} \{f(x) := \mathbb{E}[F(x, \xi)]\}, \tag{1.5}$$

where ξ is a random vector whose probability distribution P is supported on set $\Xi \subset \mathbb{R}^d$ and $F : X \times \Xi \to \mathbb{R}$. We assume that for every $\xi \in \Xi$, the function $F(\cdot, \xi)$ is convex on X, and that the expectation

$$\mathbb{E}[F(x, \xi)] = \int_{\Xi} F(x, \xi) dP(\xi) \tag{1.6}$$

is well defined and finite valued for every $x \in X$. It follows that the function $f(\cdot)$ is convex and finite valued on X. Moreover, we assume that f is continuous on X. With these assumptions, (1.5) becomes a convex programming problem.

A difficulty of solving stochastic convex problem (1.5) is that the objective is written as an expectation function for which exact zeroth and first-order oracles may not exist. Moreover, evaluating integral in (1.6) cannot be computed efficiently to the required accuracy for high dimension d. Hence, a common notion is to assume existence of stochastic oracle $\mathcal{SO}$, which we describe next. At iteration t of the algorithm, $x_t \in X$ being the input, the $\mathcal{SO}$ outputs a vector $G(x_t, \xi_t)$, where $\{\xi_t\}_{t \geqslant 1}$ is a sequence of i.i.d. random variables (also independent of search points x_t) whose probability distribution P is supported on $\Xi \subseteq \mathbb{R}^d$. Following assumptions are made on Borel functions $G(x, \xi_t)$.

For any $x \in X$, we have

$$\mathbb{E}[G(x, \xi_t)] = g(x) \in \partial f(x),$$
$$\mathbb{E}\big[\|G(x, \xi_t) - g(x)\|_*^2\big] \leqslant \sigma^2,$$

where $\partial f(x)$ denotes the subdifferential of f at x. Note that we assume that we can obtain an unbiased estimator of the subgradient whose second moment is uniformly bounded.

There exist two competing approaches for solving (1.5): *stochastic approximation* (SA) and

sample average approximation (SAA), both of which have a long history. Given the vast amount of literature, we focus on just one work which is relevant for our discussion here. Recently, [77] demonstrated that a properly modified SA method with iterate averaging can be competitive and even outperform the SAA approach for a certain class of stochastic problems. Moreover, this algorithm exhibits unimprovable rate of convergence $\mathbb{E}[f(x_N) - f^*] \leqslant O\Big(\frac{M+\sigma}{\sqrt{N}}\Big)$ where M is the modulus of Lipschitz continuity of f. Note that the term $\frac{M}{\sqrt{N}}$ inside the convergence rate is equivalent to $\frac{M^2}{\varepsilon^2}$ upper bound on the complexity $\mathrm{Compx}(\varepsilon)$.

In the last section, we briefly discussed that the subgradient method converges optimally for general nonsmooth problems however has suboptimal dependence on Lipschitz constant for Lipschitz smooth component of the composite optimization problem. In the next subsection, we describe another method that exhibits unified and optimal convergence complexity for both smooth and nonsmooth components which can be stochastic as well. Such unified complexity results show the benefits of a systematic study of complexity analysis. Indeed the search for methods with faster convergence such as Nesterov's optimal method for smooth convex optimization or unified complexity results in the upcoming section was motivated by the lower bound on the complexity as described in Nemirovski and Yudin [79].

1.3.1 Unified method for stochastic composite convex optimization

Here, we consider the composite optimization problem (1.3) along with stochastic first-order oracle information for function ψ satisfying the aforementioned assumptions of $\mathcal{SO}$. This problem is referred to as a stochastic composite optimization problem.

In the following, we describe Lan's accelerated stochastic approximation (AC-SA) [57] algorithm which exhibits unified and optimal convergence complexity for stochastic composite optimization problem. Note that the complexity for stochastic composite optimization can be lower bounded by

$$\mathrm{Compx}(\varepsilon) \geqslant \left(\frac{L}{\varepsilon}\right)^{1/2} + \frac{M^2 + \sigma^2}{\varepsilon^2},$$

where we change the notion of accuracy in (1.2) to the expectation notion

$$\varepsilon(x, \mathcal{I}) := \mathbb{E}f(x) - f^*,$$

where the expectation is taken over x, assumed to be the output of a stochastic algorithm. Note that the expected optimality gap is a natural criterion for error in stochastic convex optimization since the solution output by a stochastic algorithm is essentially a random variable. For the time being, we also ignore the function constraints in the definition of accuracy. AC-SA method achieved an upper bound on the complexity which is at most a constant factor worse than the lower bound mentioned above.

AC-SA method is motivated by two different algorithms that were developed separately for solving two different classes of problems. The first inspiration comes from Mirror Descent SA which is optimal for nonsmooth and stochastic convex optimization, and secondly from Nesterov's accelerated method which is optimal for smooth convex optimization. Without further ado, let us see the AC-SA algorithm.

Accelerated Stochastic Approximation (AC-SA) method:

0. Let $x_1^{ag} = x_1 \in X$. Set $t = 1$.

1. $x_t^{md} = \frac{2}{t+1}x_t + \frac{t-1}{t+1}x_t^{ag}$.

2. Call $\mathcal{SO}$ to compute $G(x_t^{md}, \xi_t)$. Compute $(x_{t+1}, x_{t+1}^{ag}) \in X \times X$ as

$$x_{t+1} = \operatorname*{argmin}_{x \in X} \frac{2}{t+1}\gamma\langle G(x_t^{md}, \xi_t), x\rangle + \frac{1}{2}\|x - x_t\|_2^2,$$
$$x_{t+1}^{ag} = \frac{2}{t+1}x_{t+1} + \frac{t-1}{t+1}x_t^{ag}.$$

3. Set $t \leftarrow t + 1$ and go to step 1.

The main convergence result for AC-SA method is the following:

Theorem 1.3.1 *Suppose γ in AC-SA algorithm is set to $\gamma = \min\left\{\frac{1}{2L}, \frac{\sqrt{6}D_X}{(N+2)^{3/2}(4M^2+\sigma^2)^{1/2}}\right\}$, where $D_X = \max_{x,y\in X}\|x-y\|_2$ and N is a fixed in advance number of iterations. Then, we have*

$$\mathbb{E}[\psi(x_{N+1}^{ag}) - \psi^*] \leqslant \frac{4LD_X^2}{N(N+2)} + \frac{4D_X\sqrt{4M^2+\sigma^2}}{\sqrt{N}}.$$

It is not difficult to observe that the upper bound on the complexity, i.e., N_ε for obtaining $\mathbb{E}[\psi(x_{N+1}^{ag}) - \psi^*] \leqslant \varepsilon$ is at most

$$\text{Compx}(\varepsilon) \leqslant N_\varepsilon \leqslant O(1)\left\{\left(\frac{L}{\varepsilon}\right)^{1/2} + \frac{M^2+\sigma^2}{\varepsilon^2}\right\}.$$

Note that the problem addressed by AC-SA does not contain function constraint. Traditionally most studies on function constrained optimization (with a possibly nonconvex objective and constraints) were focused on obtaining an asymptotic convergence result. We will look at this quite general case in Section 1.5. However, there are some convergence results for convex function constrained optimization which we will discuss briefly in the next section.

1.4 Advances in Convex Function Constrained Optimization

There are various methods for solving convex function constrained optimization with provable convergence guarantees. We divided them into three separate categories.

First, there are primal methods that do not involve Lagrange multipliers of the constraints functions. A notable example of this category is the level-set method due to Lemaréchal et al. [63, 84] which considers cases of nonsmooth and smooth deterministic function constrained optimization problem separately. More recently, [67] extended level-set method for nonsmooth stochastic problems. Another type of primal method includes the cooperative subgradient method that was first introduced by Polyak [90] and later extended for stochastic problems in [62]. Note that in both a and b, the stochastic oracle requires a subgaussian tail which is a quite restrictive assumption than the bounded second-moment oracle used in AC-SA. Moreover, these primal methods don't make the best use of smooth components of the objective/constraints and hence cannot achieve a unified complexity result like AC-SA with accelerated convergence for the smooth part. This problem re-

sembles that of the subgradient method which is optimal for nonsmooth optimization problems but has worse than optimal dependence on Lipschitz constant when a smooth component is present.

The second category consists of augmented Lagrangian and penalty methods. The first non-asymptotic convergence result for these methods was shown in a series of papers [55, 56] for linear constraints and general convex objective. The linearity assumption on constraints was relaxed in [114]. However, all of these methods deal with smooth deterministic optimization problems, and hence, the problem class is quite restrictive.

The third category consists of primal-dual methods. Here, the constrained optimization problem is converted into an equivalent saddle point reformulation and is solved using primal-dual methods such as mirror-prox [78] or a recent primal-dual method proposed in [46]. In particular, for problem (1.1), the Lagrangian saddle point reformulation has the following form:

$$\min_{x \in X} \max_{y \geqslant \mathbf{0}} \left\{ \mathcal{L}(x, y) := f_0(x) + \textstyle\sum_{i=1}^{m} y_i f_i(x) \right\}. \tag{1.7}$$

The main challenge in the use of these algorithms is that they may not converge directly for the saddle point formulation in (1.7) since the domain of the dual variable, y, is unbounded. In particular, for general convex-concave saddle point problem, primal-dual type algorithms converge under the assumption that

$$\|\nabla_x \mathcal{L}(x_1, y) - \nabla_x \mathcal{L}(x_2, y)\|_* \leqslant L\|x_1 - x_2\|,$$

for all $x_1, x_2 \in X$ and $y \geqslant \mathbf{0}$. Since the domain of y is unbounded, a constant L satisfying the uniform upper bound above does not exist for saddle point problem (1.7) with a nonlinear convex function $f_i, i \in [m]$. Hence, primal-dual method requires bounding the dual feasible set such that at least one optimal dual solution of (1.7) is contained sufficiently inside that set. In general, it may not be possible to find a working bound a priori. Moreover, primal-dual methods are not known to converge for nonsmooth or stochastic function constrained optimization problems.

In Chapter 3, we will see a primal-dual method that modifies an existing algorithm slightly and answers quite a few open problems regarding convex composite function constrained optimization

problem. It gives a unified complexity result, a first for (1.1) with accelerated convergence for Lipschitz smooth components. It also gives convergence under bounded second moment oracle for stochastic component and its convergence rate on the stochastic component is optimal. Moreover, this method does not require the aforementioned boundedness of the dual feasible set and can directly deal with (1.7) as well as nonsmooth problems. We will see a more elaborate discussion in Chapter 3.

Another closely related problem is when instead of the dual feasible set, the primal feasible set is quite large. This problem arises in certain linear programs associated with fundamental problems in combinatorial optimization. In particular, when primal set X in (1.7) is an ℓ_∞ ball, then the diameter of this set cannot be ignored. For such problems, even though convergence can be obtained using a standard method, the diameter of ℓ_∞ ball adds another $\sqrt{n}$ factor, where n is the dimension of LP. This can be a huge factor for most LPs of practical interest. Here, we need more specialized attention to deal with this well-known ℓ_∞-barrier. We will look at this problem in more detail in Chapter 6.

For now, we shift our focus back to the brief overview of nonconvex optimization.

1.5 Advances in Composite Nonconvex Optimization

In this section, we consider the following composite optimization problem:

$$\begin{aligned} \min_{x \in X} \quad & \psi_0(x) := f_0(x) + \chi_0(x) \\ \text{s.t.} \quad & \psi_i(x) := f_i(x) + \chi_i(x) \leqslant 0, \quad i = 1, \ldots, m, \end{aligned} \tag{1.8}$$

where $f_0 : X \to \mathbb{R}$ and $f_i : X \to \mathbb{R}$, $i = 1, \ldots, m$ are continuous functions which are not necessarily convex but satisfy that gradients are Lipschitz continuous and $\chi_i : X \to \mathbb{R}$ are convex, possibly nonsmooth functions.

The past few years have also seen a resurgence of interest in the design of efficient algorithms for nonconvex stochastic optimization, especially for stochastic and finite-sum problems due to

their importance in machine learning. Most of these studies need to assume that the constraints are convex, and focus on the analysis of iteration complexity, i.e., the number of iterations required to find an approximate stationary point, as well as possible ways to accelerate such approximate solutions.

If the nonconvex function constraints do not appear, one type of approach for solving (4.1) is to directly generalize stochastic gradient descent type methods (see [39, 41, 93, 1, 36, 123, 109, 123, 109, 88, 54]) for solving problems with nonconvex objective functions. An alternative approach is to indirectly utilize convex optimization methods within the framework of proximal-point methods which transfer nonconvex optimization problems into a series of convex ones (see [45, 13, 37, 27, 51, 60, 91, 85]). While direct methods are simpler and hence easier to implement, indirect methods may provide stronger theoretical performance guarantees under certain circumstances, e.g., when the problem has a large conditional number, many components and/or multiple blocks [60].

However, if nonconvex function constraints $\psi_i(x) \leqslant 0$ do appear in (4.1), the study on its solution methods is scarce. While there is a large body of work on the asymptotic analysis and the optimality conditions of penalty-based approaches for general constrained nonlinear programming (for example, see [12, 74, 4, 3, 30]), only a few works discussed the complexity of these methods for solving problems with nonconvex function constraints [21, 108, 34]. However, these techniques do not apply to our setting because they cannot guarantee the feasibility of the generated solutions, but certain local non-increasing properties for the constraint functions. On the other hand, the feasibility of the nonconvex function constraints appears to be important in certain problems of interest.

In chapter 4, we will see some new algorithm for nonconvex algorithm. We will show asymptotic as well as the rate of convergence results of this algorithm to a KKT-point. In order to talk about KKT-condition, we will also introduce a subdifferential for nonsmooth nonconvex problem (1.8). We analyze the convergence result under various constraint qualifications. The details of this algorithm are a bit involved so we will discuss them in more detail in Chapter 4.

1.6 Organization of the Thesis

The thesis is organized as follows.

In Chapter 2, we explore some basic questions on the complexity of training neural networks with ReLU activation function. We show that it is NP-hard to train a two-hidden layer feedforward ReLU neural network. If the dimension of the input data and the network topology is fixed then we show that there exists a polynomial-time algorithm for the same training problem. We also show that if sufficient over-parameterization is provided in the first hidden layer of ReLU neural network then there is a polynomial-time algorithm that finds weights such that output of the over-parameterized ReLU neural network matches with the output of the given data.

In Chapter 3, we present a novel Constraint Extrapolation (ConEx) method for solving convex function constrained problems, which utilizes linear approximations of the constraint functions to define the extrapolation (or acceleration) step. We show that this method is a unified algorithm that achieves the best-known rate of convergence for solving different function constrained convex composite problems, including convex or strongly convex, and smooth or nonsmooth problems with a stochastic objective and/or stochastic constraints. Many of these rates of convergence were in fact obtained for the first time in the literature. Besides, ConEx is a single-loop algorithm that does not involve any penalty subproblems. Contrary to existing primal-dual methods, it does not require the projection of Lagrangian multipliers onto a (possibly unknown) bounded set.

In Chapter 4, we study the nonconvex function constrained optimization problem. We first introduce a new proximal point method which transforms the initial nonconvex problem into a sequence of convex function constrained subproblems. We establish the convergence and rate of convergence of this algorithm to KKT points under different constraint qualifications. For practical use, we present inexact variants of this algorithm, in which approximate solutions of the subproblems are computed using the aforementioned ConEx method and establish their associated rate of convergence.

In Chapter 5, we study a constrained model for inducing sparsity. This model consists of a

general convex or nonconvex objective and a variety of continuous nonconvex (and nonsmooth) sparsity-inducing constraints. For this constrained model, we propose a novel proximal point algorithm that solves a sequence of convex subproblems with gradually relaxed constraint levels. Each subproblem, having a proximal point objective and a convex surrogate constraint, can be efficiently solved based on a fast routine for projection onto the surrogate constraint. We establish the asymptotic convergence of the proposed algorithm to the Karush-Kuhn-Tucker (KKT) solutions. We also establish new convergence complexities to achieve an approximate KKT solution when the objective can be smooth/nonsmooth, deterministic/stochastic, and convex/nonconvex with the complexity that is on a par with gradient descent when applied to nonconvex regularized problems. To the best of our knowledge, this is the first study of the first-order methods with complexity guarantee for nonconvex sparse-constrained problems. We perform numerical experiments to demonstrate the effectiveness of our new model and the efficiency of the proposed algorithm for large scale problems.

In Chapter 6, we give a faster width-dependent algorithm for mixed packing-covering LPs. Mixed packing-covering LPs are fundamental to combinatorial optimization in computer science and operations research. Our algorithm finds a $1+\varepsilon$ approximate solution in time $O(Nw/\varepsilon)$, where N is number of nonzero entries in the constraint matrix, and w is the maximum number of nonzeros in any constraint. This algorithm is faster than Nesterov's smoothing algorithm which requires $O(N\sqrt{n}w/\varepsilon)$ time, where n is the dimension of the problem. The current best width-independent algorithm for this problem runs in time $O(N/\varepsilon^2)$ [116] and hence has worse running time dependence on ε. Many real life instances of mixed packing-covering problems exhibit small width and for such cases, our algorithm can report higher precision results when compared to width-independent algorithms.

CHAPTER 2

COMPLEXITY OF TRAINING RELU NEURAL NETWORK

In this chapter, we study the computational complexity of training ReLU neural networks. First, we provide a brief introduction of neural networks and ReLU neural networks.

2.1 Introduction to Neural Networks

Deep neural networks (DNNs) are functions computed on a graph parameterized by its edge weights. More formally, the graph corresponding to a DNN is defined by input and output dimensions $w_0, w_k \in \mathbb{Z}_+$, number of hidden layers $k \in \mathbb{Z}_+$, and a sequence of k natural numbers $w_1, w_2, \ldots, w_k$ representing the number of nodes in each of the hidden k-layers. The function computed on the DNN graphs is:

$$f := \tau \circ a_k \circ \cdots \circ a_2 \circ \tau \circ a_1,$$

where $\circ$ is function composition, τ is a nonlinear function (applied componentwise) called as the activation function, and $a_i : \mathbb{R}^{w_{i-1}} \to \mathbb{R}^{w_i}$ are affine functions. Given the input and corresponding output data, the problem of training a deep neural network can be thought of as determining the edge weights of the directed layered graph for which output of the neural network matches the output data as closely as possible. Formally, given a set of input and output data $\{(x^i, y^i)\}_{i=1}^N$ where $(x^i, y^i) \in \mathbb{R}^{w_0} \times \mathbb{R}^{w_k}$, and a loss function $l : \mathbb{R}^{w_k} \times \mathbb{R}^{w_k} \to \mathbb{R}_{\geqslant 0}$ (e.g., l can be the square loss function), the task is to determine the weights that define the affine function a_i's such that

$$\sum_{i=1}^{N} l(f(x^i), y^i) \tag{2.1}$$

is minimized.

Some commonly studied activation functions are: threshold function, sigmoid function and ReLU function. ReLU is one of the important activation functions used widely in applications. However, the problem of complexity of training multi-layer fully-connected ReLU neural network remained open. This is where we add our contributions. Before formally stating our results, we take a look at the current state-of-the-art in the literature.

2.2 Complexity of training neural networks

First, we provide a brief overview of the complexity results for training neural networks with threshold activation function. The threshold (sign) function is given by

$$\text{sgn}(x) := \begin{cases} 1 & \text{if } x > 0 \\ & \\ -1 & \text{if } x < 0 \end{cases}.$$

It was shown by Blum et al. [15] that the problem of training a simple two layer neural network with two nodes in the first layer and one node in the second layer while using threshold activation function at all the nodes is NP-complete. The problem turns out to be equivalent to separation by two hyperplanes which was shown to be NP-complete by Megiddo [75]. There are other hardness results such as crypto hardness for intersection of k-hyperplanes which apply to neural networks with threshold activation function [96, 50].

2.2.1 Complexity of training neural network with rectified linear unit (ReLU) activation function

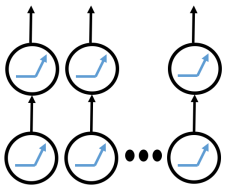
Theoretical worst case results presented above, along with limited empirical successes led to DNN's going out of favor by late 1990s. However, in recent times, DNNs became popular again due to the success of first-order gradient based heuristic algorithms for training. This success started with the work of [47] which gave an empirical evidence that if DNNs are initialized properly then we can find good solutions in reasonable runtime. This work was soon followed by series of early successes of deep learning in natural language processing [25], speech recognition [76] and visual object classification [53]. It was empirically shown by [118] that a sufficiently

over-parameterized neural network can be trained to global optimality.

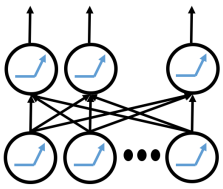
These gradient-based heuristics are not useful for neural networks with threshold activation function as there is no gradient information. Even networks with sigmoid activation function fell out of favor because gradient information is not valuable when input values are large[48]. The popular neural network architecture uses *ReLU activations* on which the gradient based methods are useful. Formally, the ReLU function is given by: $[x]_+ := \max(x, 0)$.

Related literature As discussed before, most hardness results so far are for neural networks with threshold activation function[15, 50, 96]. There are also limited results for ReLU that we discuss next: Recently, [68] examined ReLU activations from the point of view that two connected ReLU nodes, when appropriately designed, yield an approximation to threshold function. Hence training problem for such a class of ReLU network should be as hard as training a neural network with threshold activation function. Similar results are shown by [26]. In both these papers, in order to approximate the threshold activation function, the neural network studied is not a fully connected network. More specifically, in the underlying graph of such a neural network, each node in the second hidden layer is connected to exactly one distinct node in the first hidden layer, weight of the connecting edge is set to -1 with the addition of some positive bias term. Figure 2.1 shows the difference between ReLU network studied by [68, 26] and fully connected ReLU network. The architecture artificially restricts the form of the affine functions in order to prove NP-hardness. In particular, it requires connecting hidden layer matrix to be a square diagonal matrix. Due to this restriction, it was unclear whether allowing non-diagonal entries of the matrix to be non-zero would make problem easy (more parameters hence higher power to neural network function) or hard (more parameters so more things to decide).

Another line of research in understanding the hardness of training ReLU neural networks assumes that the data is coming from some distribution. More recent works in this direction include [97] which shows a smooth family of functions for which the gradient of squared error function is not informative while training neural network over Gaussian input distribution. Another study



(a) ReLU network studied in [68, 26]



(b) Fully connected ReLU

Figure 2.1: Difference between ReLU model studied in [68, 26] and typical fully connected counterpart

in this line of work considers Statistical Query (SQ) framework [101] (which contains SGD algorithms) and shows that there exists a class of special functions generated by single hidden layer neural network for which learning will require exponential number of queries (i.e. sample gradient evaluations) for the data coming from the product measure of the real valued log-concave distribution. These are interesting studies in their own right and generally consider hardness with respect to the algorithms that use stochastic gradient queries and require that such algorithm must perform minimization of the (expectation) objective functions. In comparison, we consider the framework of NP-hardness which takes into account the complete class of the polynomial time algorithms, generally assumes that the data is given and requires an optimal solution to the corresponding empirical objective.

Recently, [6] showed that a single hidden layer ReLU network can be trained in polynomial time when dimension of input, w_0, is constant.

Based on the above discussion, we see that the status of the complexity of training the multi-layer fully-connected ReLU neural network remains open. Given the importance of the ReLU NN, this is an important question. In this chapter, we take the first steps in resolving this question.

2.2.2 Our Contributions

- NP-hardness: We show that the training problem for a simple two hidden layer fully-connected NN which has two nodes in the first layer, one node in the second layer and ReLU activation function at all nodes is NP-hard (Theorem 2.4.1). Underlying graph of this network is exactly the same as that in Blum et al. [15] but all activation functions are ReLU instead

of threshold function. Techniques used in the proof are different from earlier work in the literature because there is no combinatorial interpretation to ReLU as opposed to threshold function.

- Polynomial-time solvable cases: We present two cases where the training problem with ReLU activation function can be solved in polynomial-time. The first case is when the dimension of the input is fixed (Theorem 2.4.3). This result generalizes the result from [6] and uses the hyperplane arrangement theorem for its proof.

  We also observe that when the number of nodes in the first layer of the network is equal to the number of input data points (Proposition 2.4.4) then there exists a polynomial time algorithm. The proof of this fact follows from a simple observation that reduces the problem to fitting a single hidden layer neural network and then applying the polynomial time algorithm result for single hidden layer neural network in the work of [118] This is the highly over-parameterized neural network setting. This result leads to some interesting open questions that we discuss later.

2.3 Notation and Definitions

We use the following standard set notation $[n] := \{1, \ldots, n\}$. Let $a(x) = c_1^T x + c_2$ be an affine function, then we denote a as (c_1, c_2) wherever such a notation is necessary. For any scalar α, we naturally denote affine function αa as $(\alpha c_1, \alpha c_2)$. The letter d generally denotes the dimension of input data, N denotes the number of data-points and unless explicitly specified, the output data is one dimensional.

The main training problem of interest for the paper corresponds to a neural network with 3 nodes. The underlying graph is a layered directed graph with two layers. The first layer contains two nodes and the second layer contains one node. The network is fully connected feedforward network. One can write the function corresponding to this neural network as follows:

$$F(x) = \big[w_0 + w_1\big[a_1(x)\big]_+ + w_2\big[a_2(x)\big]_+\big]_+, \tag{2.2}$$

where $a_i : \mathbb{R}^d \to \mathbb{R}$ for $i \in \{1, 2\}$ are real valued affine functions, and $w_0, w_1, w_2 \in \mathbb{R}$. The

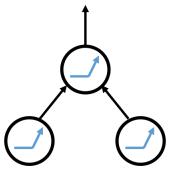


Figure 2.2: (2,1)-ReLU Neural Network. Also called 2-ReLU NN after dropping '1'. Here ReLU function is presented in each node to specify the type of activation function at the output of each node.

output of the two affine maps a_1, a_2 are the inputs to the two ReLU nodes in first hidden layer of network. The weights $\{w_0, w_1, w_2\}$ denote affine map for ReLU node in second layer. We refer to the network defined in (2.2) as (2,1)-ReLU Neural Network(NN). As its name suggests, it has 2 ReLU nodes in first layer and 1 ReLU node in second layer.

We will refer to (k, j)-ReLU NN as a generalization of $(2, 1)$-ReLU NN where there are k ReLU nodes in first layer and j ReLU nodes in second layer. Note that the output of (k, j)-ReLU NN lies in $\mathbb{R}^j$.

If there is only one node in the second layer, we will often drop the "1" and refer it as a 2-ReLU NN or k-ReLU NN depending on whether there are 2 or k nodes in the first layer, respectively. Figure 2.2 shows 2-ReLU NN.

Observation 2.3.1 *Note that*

$$w[ax + b]_+ \equiv sgn(w)[|w|(ax + b)]_+ = sgn(w)[\tilde{a}x + \tilde{b}],$$

so without loss of generality we will assume $w_1, w_2 \in \{-1, 1\}$ *in (2.2).*

Now we formally state the definition of the decision version of the training problem.

Definition 2.3.1 (Decision-version of the training problem) *Given a set of training data* $(x^i, y^i) \in \mathbb{R}^d \times \{1, 0\}$ *for* $i \in S$*, do there exist edge weights so that the resulting function* F *satisfies* $F(x^i) = y^i$ *for* $i \in S$.

The decision version of the training problem in Definition 2.3.1 is asking if it is possible to find edge weights to obtain zero loss function value in the expression (2.1), assuming l is a norm i.e. $l(a, b) = 0$ iff $a = b$.

2.4 Main Results

Theorem 2.4.1 *It is NP-hard to solve the training problem for 2-ReLU NN.*

An immediate corollary of Theorem 2.4.1 is the following:

Corollary 2.4.2 *Training problem of (2,j)-ReLU NN is NP hard, for all* $j \geqslant 1$.

The proof of Theorem 2.4.1 is obtained by reducing the 2-Hyperplane Separability Problem to the training problem of 2-ReLU NN. Details of this reduction and the proof of Theorem 2.4.1 and Corollary 2.4.2 are presented in Section 2.5.

After this work was finished, two more studies [73, 28] considered the computational complexity of training a single ReLU node and proved that it is a NP-hard problem. [73] also showed that it is NP-hard to train one hidden layer neural network with two nodes and ReLU activation at each node. This network basically removes the second layer ReLU activation and affine constant w_0 in (2.2) so that neural network function of their case can be rewritten as $F(x) = w_1\big[a_1(x)\big]_+ + w_2\big[a_2(x)\big]_+$. These are different network architectures and hence hardness of training any one of them does not necessarily imply hardness of training for remaining neural networks.

Megiddo [75] shows that the separability with fixed number of hyperplanes (generalization of 2-hyperplane separability problem) can be solved in polynomial-time in fixed dimension. Therefore 2-hyperplane separability problem can be solved in polynomial time given dimension is constant. Based on the reduction used to prove Theorem 2.4.1 , a natural question to ask is "Can one solve the training problem of 2-ReLU NN problem in polynomial time under the same assumption?". We answer this question in the affirmative.

Theorem 2.4.3 *Under the assumption that the dimension of input, d and the number of nodes in the first layer, k, are constant, then there exists a poly(N)-time solution to the training problem of k-ReLU neural network, where N is the number of data-points.*

The high-level idea of the proof is the following: each data point "passes through" the three ReLU nodes and the activation function in these nodes is "turned on" or "turned off" (i.e., the output is 0 or not). We will enumerate all possible combinations of the data points being turned on or not, which we show is poly(N) assuming d and k is fixed (by use of the Hyperplane Arrangement Theorem). Then we show that for each of these combinations and for each possible sign pattern of the weights defining the affine function applied at the second layer, corresponding optimal affine functions can be calculated via solving one convex program of poly size. Finally, we select the best optimal affine function which minimizes the loss function. Technique of Hyperplane Arrangement Theorem to enumerate partition was used in [6] for proving poly(N)-time algorithms for single hidden layer neural networks. We extend this result for k-ReLU neural network which is a two hidden layer network. The complication due to second layer ReLU node are handled by solving a convex program of poly size. We show the precise proof of Theorem 2.4.3 in Section 2.7.1. We also study this problem under over-parameterization. Structural understanding of 2-ReLU NN yields an easy algorithm to solve training problem for N-ReLU neural network over N data points. In fact, the problem can be easily reduced to a single hidden layer NN.

Proposition 2.4.4 *Given data, $\{x^i, y^i\}_{i\in[N]}$ (where we assume that x^is are distinct), then the training problem for N-ReLU NN has a poly(N,d)-time randomized algorithm, where N is the number of data-points and d is the dimension of input.*

Proof of this proposition first reduces the problem to training a single hidden layer network with N nodes on dataset of size N. Then applies polynomial time algorithm for interpolating the data from [118]. The precise details are in Section 2.7.2.

2.5 Training 2-ReLU NN is NP-hard

In this section we give details about the NP-hardness reduction for the training problem of 2-ReLU NN. We begin with the formal definition of 2-Hyperplane Separability Problem.

Definition 2.5.1 (2-Hyperplane Separability Problem) *Given a set of points $\{x^i\}_{i\in[N]} \in \mathbb{R}^d$ and a partition of $[N]$ into two sets: S_1, S_0, (i.e. $S_1 \cap S_0 = \varnothing$, $S_1 \cup S_0 = [N]$) decide whether there exist two hyperplanes $H_1 = \{x : \alpha_1^T x + \beta_1 = 0\}$ and $H_2 = \{x : \alpha_2^T x + \beta_2 = 0\}$ where $\alpha_1, \alpha_2 \in \mathbb{R}^d$ and $\beta_1, \beta_2 \in \mathbb{R}$ that separate the set of points in the following fashion:*

1. *For each point x^i such that $i \in S_1$, both $\alpha_1^T x^i + \beta_1 > 0$ and $\alpha_2^T x^i + \beta_2 > 0$.*
2. *For each point x^i such that $i \in S_0$, $\alpha_1^T x^i + \beta_1 < 0$ or $\alpha_2^T x^i + \beta_2 < 0$.*

The 2-hyperplane separability problem is NP-complete [75]. Note the difference between conditions 1 and 2 above. First one is an "AND" statement and second is an "OR" statement. Geometrically, solving 2-hyperplane separability problem means that finding two affine hyperplanes $\{\alpha_1, \beta_1\}$ and $\{\alpha_2, \beta_2\}$ such that all points in set S_1 lie in one quadrant formed by two hyperplanes and all points in set S_0 lie outside that quadrant. Due to this geometric intuition, the problem is called separation by 2-hyperplane separability. We will construct a polynomial reduction from this NP-complete problem to training 2-ReLU NN, which will prove that training 2-ReLU NN is NP-hard.

Remark 2.5.1 (Variants of 2-hyperplane separability) *Note here that some sources also define 2-hyperplane separability problem with minor difference. In particular, the change is that strict inequalities, '$>$', in Definition 2.5.1.1 are diluted to inequalities, '$\geqslant$'. In fact, these two problems are equivalent in the sense that there is a solution for the first problem if and only if there is a solution for the second problem. Solution for the first problem implies solution for the second problem trivially. Suppose there is a solution for the second problem, that implies there exist $\{\alpha_1, \beta_1\}$ and $\{\alpha_2, \beta_2\}$ such that for all $i \in S_0$ we have either $\alpha_1^T x^i + \beta_1 < 0$ or $\alpha_2^T x^i + \beta_2 < 0$.*

This implies $\epsilon := \min_{i \in S_0} \max\{-\alpha_1 x^i - \beta_1, -\alpha_2 x^i - \beta_2\} > 0$. *So if we shift both planes by* $\frac{1}{2}\epsilon$ *i.e.* $\beta_i \leftarrow \beta_i + \frac{1}{2}\epsilon$ *then this is a solution to the first problem.*

Assumption: $\mathbf{0} \in S_1$ (Here $\mathbf{0} \in \mathbb{R}^d$ is a vector of zeros.) Suppose we are given a generic instance of 2-hyperplane separability problem with data-points $\{x^i\}_{i \in [N]}$ from $\mathbb{R}^d$ and partition S_1 and S_0 of the set $[N]$. Since the answer of 2-hyperplane separability instance is invariant under coordinate translation, we shift the origin to any x^i for $i \in S_1$, and therefore assume that the origin belongs to S_1 henceforth.

2.5.1 Reduction

Now we create a particular instance for 2-ReLU NN problem from a general instance of 2-hyperplane separability. We add two new dimensions to each data-point x^i. We also create a label, y^i, for each data-point. Moreover, we add a constant number of extra points to the training problem. Exact details are as follows:

Consider training set $\{(x^i, 0, 0), y^i\}_{i \in [N]}$ where $y^i = \begin{cases} 1 & \text{if } i \in S_1 \\ 0 & \text{if } i \in S_0 \end{cases}$.

Add additional 18 data points to the above training set as follows:

$\{p_1 \equiv \{(\mathbf{0}, 1, 1), 1\}, p_2 \equiv \{(\mathbf{0}, 2, 1), 1\}, p_3 \equiv \{(\mathbf{0}, 1, 2), 1\}, p_4 \equiv \{(\mathbf{0}, 2, 2), 1\},$

$p_5 \equiv \{(\mathbf{0}, 0.75, 1.5), 1\}, p_6 \equiv \{(\mathbf{0}, 2.25, 1.5), 1\}, p_7 \equiv \{(\mathbf{0}, 1.5, 0.75), 1\}, p_8 \equiv \{(\mathbf{0}, 1.5, 2.25), 1\},$

$p_9 \equiv \{(\mathbf{0}, 1, -1), 0\}, p_{10} \equiv \{(\mathbf{0}, 2, -1), 0\}, p_{11} \equiv \{(\mathbf{0}, 3, -1), 0\},$

$p_{12} \equiv \{(\mathbf{0}, -1, 1), 0\}, p_{13} \equiv \{(\mathbf{0}, -1, 2), 0\}, p_{14} \equiv \{(\mathbf{0}, -1, 3), 0\},$

$p_{15} \equiv \{(\mathbf{0}, -1, 0), 0\}, p_{16} \equiv \{(\mathbf{0}, 0, -1), 0\}\},$

$p_{17} \equiv \{(\mathbf{0}, -1, 5), 0\}, p_{18} \equiv \{(\mathbf{0}, 5, -1), 0\}\}.$

Let's call the set of additional data points with label 1 as T_1 and additional data points with label 0 as T_0. These additional data points (we refer to these points as the "gadget points") are of fixed size. So this is a polynomial time reduction.

Figure 2.3 shows the gadget points. Note that origin is added to the gadget because there exists $i \in S_1$ such that $x^i = \mathbf{0}$. Hence training set has the data-point $\{(\mathbf{0}, 0, 0), 1\}$.

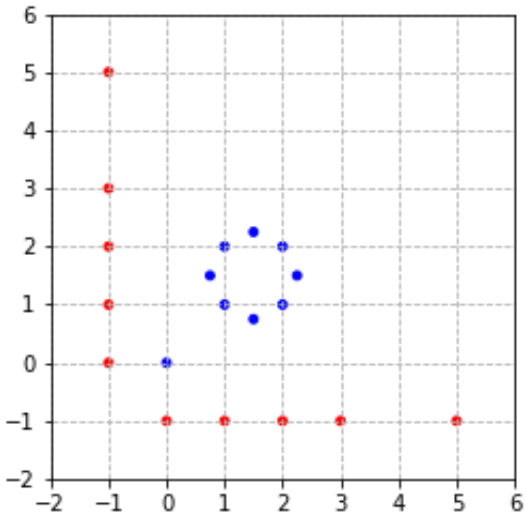


Figure 2.3: Gadget: Blue points represent set T_1 and red points represent set T_0.

Let's call the training problem of fitting 2-ReLU NN to this data as (**P**). In the context of the training problem (**P**), we abuse the notation and call the set of points $(x^i, 0, 0)$ with label 1 as S_1 and the set of points $(x^i, 0, 0)$ with label 0 as S_0. In particular, there is a direct correspondence between the sets S_1, S_0 defined in 2-hyperplane separability problem and sets S_1, S_0 defined for 2-ReLU NN training problem (**P**). Use of our notation is generally clear from the context.

Now what remains is to show that the general instance of 2-hyperplane separability has a solution if and only if the constructed instance of 2-ReLU NN has a solution. In order to understand our approach better, we introduce the notion of "hard-sorting". Hard-sorting is formally defined below, and its significance is stated in Lemma 2.5.5.

Definition 2.5.2 (Hard-sorting) *We say that a set of points $\{\pi^i\}_{i \in S}$, partitioned into two sets Π_0, Π_1 can be hard-sorted with respect to Π_1 if there exist two affine transformations l_1, l_2 and scalars w_1, w_2, c such that the following condition is satisfied:*

$$w_1\big[l_1(\pi)\big]_+ + w_2\big[l_2(\pi)\big]_+ \begin{cases} = c & \text{for all } \pi \in \Pi_1 \\ < c & \text{for all } \pi \in \Pi_0 \end{cases} \tag{2.3}$$

Being able to hard-sort implies that after passing the data through two nodes of the first hidden layer, the scalar input to the second hidden layer node must have a separation of the data-points in Π_1 and the data-points in Π_0, moreover, scalar input for all data points in Π_1 must be a constant.

Remark 2.5.2 *If there exists scalars w_1, w_2, c and affine transformations l_1, l_2 such that*

$$w_1\big[l_1(\pi)\big]_+ + w_2\big[l_2(\pi)\big]_+ = \begin{cases} = c & \textit{for all } \pi \in \Pi_1 \\ > c & \textit{for all } \pi \in \Pi_0. \end{cases}$$

then $-w_1, -w_2, -c, l_1, l_2$ satisfy condition (2.3) *of hard-sorting.*

Remark 2.5.3 *Let $\overline{\Pi}_0 \subset \Pi_0$ and $\overline{\Pi}_1 \subset \Pi_1$. Then hard-sorting of $\Pi_0 \cup \Pi_1$ with respect to $\Pi_1 \Rightarrow$ hard-sorting of $\overline{\Pi}_0 \cup \overline{\Pi}_1$ with respect to $\overline{\Pi}_1$.*

Remark 2.5.4 *Without loss of generality, we may assume that $w_1, w_2 \in \{-1, 1\}$.*

It is not difficult to see that hard-sorting implies (**P**) has a solution. We show that hard-sorting is also required for solving training problem. This is formally stated in lemma below.

Lemma 2.5.5 *The 2-ReLU NN training problem (**P**) has a solution if and only if data-points $S_1 \cup T_1 \cup S_0 \cup T_0$ are hard-sorted with respect to $S_1 \cup T_1$.*

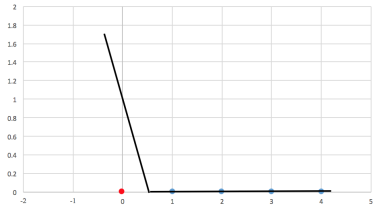
The proof of Lemma 2.5.5 can be found in Section 2.7.4 .

Figure 2.4 below explains geometric interpretation of Lemma 2.5.5 We use the hard-sorting characterization of the solution of the training problem (**P**) extensively. We first show the forward direction of the reduction in the lemma below. This is also the easier direction.

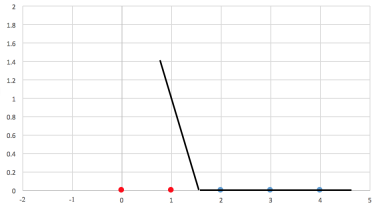
Lemma 2.5.6 *If 2-hyperplane separability problem has a solution then problem (**P**) has a solution.*

The proof of Lemma 2.5.6 can be found in Section 2.7.3.

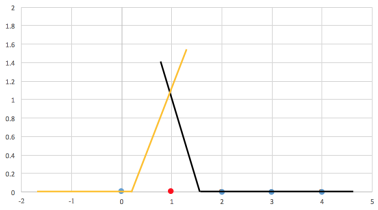
To prove reverse direction we need to show that if a set of weights solve the training problem (**P**) then we can generate a solution to the 2-hyperplane separability problem. In the rest of the proof



(a) Input is hard-sorted. This can give a perfect fit.



(b) Since there are two red points so input is not hard-sorted. This cannot give a perfect fit.



(c) Since blue points lies on different side of red points so input is not hard-sorted. This cannot give a perfect fit.

Figure 2.4: X-axis in figures above is output of the first layer of 2-ReLU NN i.e. $w_1\big[l_1(\pi)\big]_+ + w_2\big[l_2(\pi)\big]_+$. Y-axis is the output of second hidden layer node. Since output of first hidden layer goes to input of second hidden layer, we are essentially trying to fit ReLU node of second hidden layer. In particular, red and blue dots represent output of first hidden layer on data points with label 1 and 0 respectively. In fig (a) we see that hard-sorted input can be classified as $0/1$ by a ReLU function. In fig (b) and (c) we see that input which is not hard-sorted cannot be classified exactly as $0/1$ by a ReLU function.

we will argue that the only way to solve the training problem (**P**) for 2-ReLU NN or equivalently hard-sort data-points is to find two affine function $a_1, a_2 : \mathbb{R}^{d+2} \to \mathbb{R}$ such that i) $a_1(x) \leqslant 0$ and $a_2(x) \leqslant 0$ for all $x \in S_1 \cup T_1$ and ii) $a_1(x) > 0$ or $a_2(x) > 0$ for all $x \in S_0 \cup T_0$. If such a solution exists then there exists a solution to 2-hyperplane separability problem after dropping coefficients of last two dimensions of affine functions $-a_1$ and $-a_2$. Note that changing '$<$' to '$\leqslant$' in 2-affine separability problem is valid in view of Remark 2.5.1.

We will first show that we can hard-sort the gadget points only under the properties of a_1 and a_2 mentioned above. This implies that a solution to (**P**) which hard-sorts all points (including the gadget points) must have same properties of a_1 and a_2. This follows from counter-positive of Remark 2.5.3 i.e. if subset of data-points cannot be hard-sorted then all data-points cannot be hard-sorted. Henceforth, we will focus on the gadget data-points (or the last two dimensions of the data).

Gadget Points and Hard-Sorting

In the following lemma, we show a necessary condition on a_1, a_2 satisfying hard-sorting of gadget data points $T_1 \cup T_0 \cup \{\mathbf{0}\}$ with respect to $T_1 \cup \{\mathbf{0}\}$.

Lemma 2.5.7 *Suppose affine functions $a_1, a_2 : \mathbb{R}^{d+2} \to \mathbb{R}$ and scalars w_1, w_2, c satisfy hard-sorting of the data-points $T_1 \cup T_0 \cup \{\mathbf{0}\}$ with respect to $T_1 \cup \{\mathbf{0}\}$ then all points in T_1 must satisfy $a_1(x) \leqslant 0$, $a_2(x) \leqslant 0$. Moreover, we must have $w_1 = w_2 = -1$ and $c = 0$.*

Note that in view of Lemma 2.5.7 and counter-positive of Remark 2.5.3, we have that affine function $a_1, a_2 : \mathbb{R}^{d+2} \to \mathbb{R}$ and scalars w_1, w_2, c satisfying hard-sorting of $S_1 \cup T_1 \cup S_0 \cup T_0$ with respect to $S_1 \cup T_1$ must satisfy

$$-\big[a_1(x)\big]_+ - \big[a_2(x)\big]_+ \begin{cases} = 0 & \text{if } x \in S_1; \\ < 0 & \text{if } x \in S_0 \end{cases}.$$

The above condition is equivalent to the requirement that $a_1(x) \leqslant 0,\ a_2(x) \leqslant 0$ for all $x \in S_1$ and $a_1(x) > 0$ or $a_2(x) > 0$ for $x \in S_0$. After dropping the last two dimensions of $-a_1$ and $-a_2$, we obtain the solution for 2-affine separability problem. Now that we have reduced the problem to the key lemma above, the main purpose of this section is to prove Lemma 2.5.7.

Note that for each data point in the gadget $T_1 \cup T_0\{\mathbf{0}\}$, the first d elements are always 0. So for the sake of gadget, we may assume that $a_1, a_2 : \mathbb{R}^2 \to \mathbb{R}$ and the gadgets lies in $\mathbb{R}^2$. They can be thought of as the projection of the original $a_i : \mathbb{R}^{d+2} \to \mathbb{R}$ and $\mathbf{0} \in \mathbb{R}^{d+2}$ to last two dimension which are relevant for gadget data points $T_1 \cup T_0$. Due to this observation, we assume that $a_1, a_2 : \mathbb{R}^2 \to \mathbb{R}$ henceforth for this subsection and provide a proof of Lemma 2.5.7 under this assumption.

The proof of Lemma 2.5.7 is divided into the following sequence of results.

Proposition 2.5.8 *Suppose that a_1, a_2 satisfy hard-sorting of $T_1 \cup T_0$ with respect to T_1 then there exists $x \in T_1$ such that $a_1(x) \leqslant 0,\ a_2(x) \leqslant 0$.*

Proof of Proposition 2.5.8 can be found in Section 2.7.5.

Next we show one more simple proposition which is critical in proving the final result. The proof of this proposition can be found in Section 2.7.7.

Proposition 2.5.9 *Affine functions a_1, a_2 and weights w_1, w_2 satisfy hard-sorting of $T_1 \cup T_0 \cup \{\mathbf{0}\}$ with respect to $T_1 \cup \{\mathbf{0}\}$ then w_1, w_2 **must** satisfy $w_1 = w_2 = -1$.*

We are now ready to present the prove Lemma 2.5.7.

Proof of Lemma 2.5.7. Since a_1, a_2 satisfy hard-sorting of the data points $T_1 \cup T_0 \cup \{\mathbf{0}\}$ with respect to $T_1 \cup \{\mathbf{0}\}$ then, in view of Proposition 2.5.8 and Proposition 2.5.9, we have

1. $\exists x \in T_1$ such that $a_1(x) \leqslant 0, a_2(x) \leqslant 0$.
2. $w_1 = w_2 = -1$.

Then we have that $-\big[a_1(x)\big]_+ - \big[a_2(x)\big]_+ = 0$ for all $x \in T_1$, due to condition (2.3) of hard-sorting. This implies $a_1(x) \leqslant 0, a_2(x) \leqslant 0$ for all $x \in T_1$. So we conclude the proof. □

In the next section, we show that this result on the gadget data-points gives us the solution to the original 2-hyperplane separability problem.

From Gadget Data to Complete Data

Lemma 2.5.10 *If there is a solution to the problem (**P**), then there is a solution to corresponding 2-hyperplane separability problem.*

Proof. Note that if there is a solution to problem (**P**), then by Lemma 2.5.5, we must have $a_1, a_2 : \mathbb{R}^{d+2} \to \mathbb{R}$ and w_1, w_2, c hard-sorting $S_1 \cup T_1 \cup S_0 \cup T_0$ with respect to $S_1 \cup T_1$. In view of Lemma 2.5.7 and counter-positive of Remark 2.5.3, we have

1. $w_1 = w_2 = -1$.
2. $w_1\big[a_1(x)\big]_+ + w_2\big[a_2(x)\big]_+ = 0$ for all $x \in S_1 \cup T_1$ due to requirement (2.3) of hard-sorting.

Since $w_1 = w_2 = -1$, so 2 above implies $a_1(x) \leqslant 0$ and $a_2(x) \leqslant 0$ for all $x \in S_1 \cup T_1$. Moreover, we require $a_1(x) > 0$ or $a_2(x) > 0$ for all $x \in S_0 \cup T_0$ because condition (2.3) of hard-sorting. Now as discussed earlier, $-a_1, -a_2$ after ignoring coefficients of last two dimensions will yield solution to 2-hyperplane separability problem. Hence we conclude the proof. □

Now we are ready to prove the main NP-hardness theorem.

Proof of Theorem 2.4.1. Using Lemma 2.5.6 and Lemma 2.5.10, we conclude the proof. □

Below we state an immediate corollary of Theorem 2.4.1 whose proof can be found in Section 2.7.8.

Corollary 2.5.11 *Training problem of (2,j)-ReLU NN is NP hard.*

2.6 Discussion

We showed that the problem of training 2-ReLU NN is NP-hard. Given the importance of ReLU activation function in neural networks, in our opinion, this result resolves a significant gap in understanding complexity class of the problem at hand. On the other hand, we show that the problem of training N-ReLU NN is in P. So a natural research direction is to understand the complexity status when input layer has more than 2 nodes and strictly less than N nodes. A particularly interesting question in that direction is to generalize the gadget we used for 2-ReLU NN to the case of k-ReLU NN.

2.7 Proofs of Auxiliary Results

In this section, we provide proof of all auxiliary results.

2.7.1 Proof of Theorem 2.4.3

Suppose we partition the set $[N]$ into sets Q_j and $\overline{Q}_j$ such that all points in Q_j satisfy $a_j(x) \geqslant 0$ and all points in $\overline{Q}_j$ satisfy $a_j(x) < 0$ for all $j \in [k]$. Given a set $S \subseteq [k]$, we define $T(S) := \left(\bigcap_{j\in S} Q_j\right) \cap \left(\bigcap_{j\in \bar{S}} \bar{Q}_j\right)$ where $\overline{S} = [k]\backslash S$. Let $z = (a_1, \ldots, a_k, w_0, w_1, \ldots, w_k)$. Then the objective function can be written as

$$f(z) = \sum_{S\subseteq[k]} \sum_{i\in T(S)} \left(\left[w_0 + \sum_{j\in S} w_j a_j(x^i)\right]_+ - y_i\right)^2.$$

Now we can partition $T(S)$ into sets $T(S)_1$ and $T(S)_2$ for each $S \subseteq [k], S \neq \phi$. For $T(S)_1$, the ReLU term in the objective, $w_0 + \sum_{j \in S} w_j a_j(x)$ (note that this is an affine function), is constrained to be non-negative and for $T(S)_2$ the ReLU terms is constrained to be non-positive. We need not enumerate partitions of $T(\phi)$ since ReLU terms for $T(\phi)$ do not depend on data-points. The key observation is that the partition of $T(S)$ into sets $T(S)_1$ and $T(S)_2$ is a partition due to a hyperplane.

Number of combinations: According to the Hyperplane Arrangement Theorem, given a set of points $\{x^i\}_{i \in N}$ in $\mathbb{R}^d$, the number of distinct partitions created by linear separators is $O(N^d)$. Moreover, due to [32], we can enumerate all possible partitions created by linear separators in $O(N^d)$ time. Therefore, there are a total of $O(N^{kd})$ possible combinations of $Q_j, j \in [k]$. For each such $Q_j, j \in [k]$, there are 2^k non-empty subsets $T(S) \subseteq [N]$. For each $T(S), S \neq \phi$, there are $O(|T(S)|^d) = O(N^d)$ possible ways to partition $T(S)$ into $T(S)_1$ and $T(S)_2$. So number of product combinations is $O(N^{(2^k-1)d})$. Hence there are a total of $O(N^{(kd+(2^k-1)d)})$ combinations.

Number of convex programs: By Observation 2.3.1 it suffices to check for $w_1, \ldots, w_k = \pm 1$. We will divide the optimization problem in two cases $w_0 \geqslant 0$ and $w_0 \leqslant 0$. So there are a total of 2^{k+1} convex programs for each possible combination of $Q_j, T(S)_1$ for all $S \subseteq [k]$ of the following form:

$$\min \sum_{\substack{S \subseteq [k], \\ S \neq \phi}} \left\{ \sum_{i \in T(S)_1} \left(\left(w_0 + \sum_{j \in S} w_j a_j(x^i) \right) - y_i \right)^2 + \sum_{i \in T(S)_2} (0 - y_i)^2 \right\} + \sum_{i \in T_\phi} \left(\left[w_0 \right]_+ - y_i \right)^2$$

subject to constraints

$$\begin{aligned} a_j(x^i) &\geqslant 0, \qquad \forall j,\ i \in Q_j \\ a_j(x^i) &\leqslant 0, \qquad \forall j,\ i \in \overline{Q}_j \end{aligned} \tag{2.4}$$

$$\begin{aligned} w_0 + \sum_{j \in S} w_j a_j(x^i) &\geqslant 0, \qquad \forall S \subseteq [k], S \neq \phi,\ i \in T(S)_1 \\ w_0 + \sum_{j \in S} w_j a_j(x^i) &\leqslant 0, \qquad \forall S \subseteq [k], S \neq \phi,\ i \in T(S)_2 \end{aligned} \tag{2.5}$$

Moreover, we add constraint $w_0 \geqslant 0$ or $w_0 \leqslant 0$ and change the $\big[w_0\big]_+$ term in objective with w_0 or 0 respectively. Every program has $k(d+1)+1$ variables in $a_1, \ldots, a_k, w_0$. Total number of constraints is at most $kN + N + 1$. Note that, for constraints of type (2.4), for each j, number of constraints equals $|Q_j \cup \overline{Q}_j| = N$. Hence total number of constraints of type (2.4) are kN. Similarly, for constraints of type (2.5), for each $S \subseteq [k]$, we have total of $|T(S)_1 \cup T(S)_2| = |T(S)|$ constraints. Hence total constraints of type (2.5) are $\sum\limits_{\substack{S\subseteq[k],\\ S\neq\phi}} |T(S)| \leqslant N$ (This follows due to observation that $T(S), S \subseteq [k]$ is a partition of $[N]$). One more constraint is on w_0. Hence total number of constraints is $(k+1)N + 1$. Since number of constraints and variables are poly(k, d, N) and objective is convex quadratic so we conclude that this program can be solved in poly(N, k, d) time.

Finally, the total number of convex programs to be solved is $O(2^{k+1} \cdot N^{kd+(2^k-1)d})$.

2.7.2 Proof of Proposition 2.4.4

Before proving this proposition, we state a polynomial time algorithm (Theorem 1 of [118]) for training single hidden layer neural network.

Proposition 2.7.1 *There exists a poly(N, d)-time algorithm to train a single hidden layer neural network with N nodes and ReLU activations which can represent any function on sample of size N in dimension d.*

Now we are ready to prove Proposition 2.4.4.

Note that a N-ReLU NN can be written as $c(x) = \Big[\sum\limits_{j=1}^{N} w_j\big[a_j(x)\big]_+ + w_0\Big]_+$. Suppose $y = [y^1, \ldots, y^N]^T \in \mathbb{R}^N$ be a vector of labels. We may assume that $y \geqslant \mathbf{0}$ since otherwise we can add a constant term to each label in y. Then we need to find weights $w_i, i = 0, \ldots, N$ and affine functions $a_j, j = 1, \ldots, N$ such that $c(x^i) = y^i$ for all $i \in [N]$. Since $y \geqslant \mathbf{0}$ so we have $f(x^i) = y^i$ where $f(x) = \sum\limits_{j=1}^{N} w_j\big[a_j(x)\big]_+ + w_0$. Now note that function f with $w_0 = 0$ is a single hidden layer ReLU NN used in [118]. Using the fact that number of nodes in f matches number of data points, N, then applying Proposition 2.7.1, we obtain the result.

2.7.3 Proof of Lemma 2.5.6

Suppose (α_1, β_1) and (α_2, β_2) are solution satisfying condition for 2-hyperplane separability. Note that there is a data-point $\mathbf{0} \in S_1$ so we obtain $\beta_1, \beta_2 > 0$. Without loss of generality we can assume $\beta_1 = \beta_2 = 0.5$. This is due to the fact that scaling the original solution by any positive scalar yields a valid solution. Now we show that the solution of 2-hyperplane separability problem can be used to show hard-sorting of $S_0 \cup T_0 \cup S_1 \cup T_1$ with respect to $S_1 \cup T_1$. Hence in view of Lemma 2.5.5, we obtain existence of existence of solution for problem (**P**).

Set $w_1 = w_2 = -1,\ c = 0$. Moreover, for $(x, y, z) \in \mathbb{R}^{d+2}$, consider the affine map $l_1(x, y, z) = -\alpha_1^T x - y - \beta_1$ and $l_2(x, y, z) = -\alpha_2^T x - z - \beta_2$. We claim that w_1, w_2, c, l_1, l_2 satisfy hard-sorting condition (2.3) for $S_0 \cup T_0 \cup S_1 \cup T_1$ with respect to $S_1 \cup T_1$. In particular, note that

1. For $x \in S_1$, we have

$$-\big[-\alpha_1^T x - \beta_1\big]_+ - \big[-\alpha_2^T x - \beta_2\big]_+ = 0 = c.$$

2. For $x = (\mathbf{0}, l, m) \in T_1$, we have

$$-\big[-\beta_1 - l\big]_+ - \big[-\beta_2 - m\big]_+ = 0.$$

   This follows since $\beta_1 = \beta_2 = 1/2$ and $l, m \in [0.75, 2.25]$ so the two ReLU terms inside are both zero for all $x \in T_1$.

3. For $x \in S_0$, we have

$$-\big[-\alpha_1^T x - \beta_1\big]_+ - \big[-\alpha_2^T x - \beta_2\big]_+ < 0.$$

   This follows since at least one of $\alpha_1^T x + \beta_1$ and $\alpha_2^T x + \beta_2$ is strictly negative for $x \in S_0$ as (α_1, β_1) and (α_2, β_2) are solution for 2-hyperplane separability problem.

4. For $x = (\mathbf{0}, l, m) \in T_0$, we have

$$-\big[-\beta_1 - l\big]_+ - \big[-\beta_2 - m\big]_+ < 0.$$

This follows since $\beta_1 = \beta_2 = 1/2$ and either l or m equals -1 for $x \in T_0$.

This proves hard-sorting of $S_0 \cup T_0 \cup S_1 \cup T_1$ with respect to $S_1 \cup T_1$ and hence we have the existence of solution for training problem (**P**).

2.7.4 Proof of Lemma 2.5.5

We first prove the forward direction. Suppose points are hard-sorted as required by the lemma. Then define $\varepsilon := \min_{x \in S_0 \cup T_0} -w_1\big[l_1(x)\big]_+ - w_2\big[l_2(x)\big]_+ + c$. By definition, we have $\varepsilon > 0$. Then neural network $f(x) = \frac{2}{\varepsilon}\big[w_1\big[l_1(x)\big]_+ + w_2\big[l_2(x)\big]_+ - c + \varepsilon/2\big]_+$ solves training problem. This can be easily checked from the fact that

$$w_1\big[l_1(x)\big]_+ + w_2\big[l_2(x)\big]_+ - c \begin{cases} = 0 & \text{if } x \in S_1 \cup T_1; \\ < -\varepsilon & \text{if } x \in S_0 \cup T_0, \end{cases}$$

which holds under the assumption of hard-sorting.

Now we assume that points cannot be hard-sorted and conclude that there does not exist weight assignment solving training problem of 2-ReLU NN, hence proving the backward direction. Since the points cannot be hard-sorted so there does not exist any l_1, l_2, w_1, w_2, c satisfying condition (2.3). This fact along with Remark 2.5.2 implies that for all possible weights we either have
a) $w_1\big[l_1(x)\big]_+ + w_2\big[l_2(x)\big]_+$ is not constant for all $x \in S_1 \cup T_1$ or
b) If $w_1\big[l_1(x)\big]_+ + w_2\big[l_2(x)\big]_+ = c$ for all $x \in S_1 \cup T_1$ and some constant c, then same expression evaluated on $x \in S_0 \cup T_0$ is not strictly on same side of c.

If we choose l_1, l_2, w_1, w_2, c such that a) happens, then such weights will not solve training problem as their output of 2-ReLU NN for points $p \in S_1 \cup T_1$ will be at least two distinct numbers which is an undesirable outcome. Specifically, we want $\big[w_0 + w_1\big[l_1(x)\big]_+ + w_2\big[l_2(x)\big]_+\big]_+$

to evaluate to 1 for all $x \in S_1 \cup T_1$. Hence $w_1\big[l_1(x)\big]_+ + w_2\big[l_2(x)\big]_+$ must be a constant for all $x \in S_1 \cup T_1$. This requirement is violated in case a).

If we choose l_1, l_2, w_1, w_2, c such that b) happens, then we can set w_0, θ such that $F(x) = \theta\big[w_1\big[l_1(x)\big]_+ + w_2\big[l_2(x)\big]_+ + w_0\big]_+$, $w_0 + c > 0$ and $\theta = \frac{1}{w_0+c}$. Here we introduced another parameter $\theta > 0$ in the definition of F for sake of convenience of argument but note that θ can be absorbed in the definition of l_1 and l_2 to obtain the original neural network function defined (2.2). Since not all $x \in S_0 \cup T_0$ are strictly on one side, we conclude there exist $x' \in S_0 \cup T_0$ such that $w_1\big[l_1(x')\big]_+ + w_2\big[l_2(x')\big]_+ = c' \geqslant c$ hence $F(x') := \theta\big[w_1\big[l_1(x')\big]_+ + w_2\big[l_2(x')\big]_+ + w_0\big]_+ \geqslant 1$ which is an undesirable outcome for a point with label 0.

Since all choices of l_1, l_2, w_1, w_2, c satisfy either a) or b), we conclude that there does not exist weights solving training problem of 2-ReLU NN.

2.7.5 Proof of Proposition 2.5.8

In order to prove Proposition 2.5.8, we need to prove one more technical result stated below. Proof of this new proposition is deferred to Section 2.7.6 but here we state it and proceed with the proof of Proposition 2.5.8.

Proposition 2.7.2 *Let c be an arbitrary constant. Suppose affine functions $a_1, a_2 : \mathbb{R}^2 \to \mathbb{R}$ satisfy $w_1a_1(x) + w_2a_2(x) = c$ for all $x \in \mathbb{R}^2$, then such a_1, a_2 cannot satisfy hard-sorting of the data points $T_1 \cup T_0 \cup \{\boldsymbol{0}\}$ with respect to $T_1 \cup \{\boldsymbol{0}\}$.*

Remark 2.7.3 *A key corollary of Proposition 2.7.2 is that if a_1, a_2 satisfy hard-sorting of gadget data points $T_1 \cup T_0 \cup \{\boldsymbol{0}\}$ with respect to $T_1 \cup \{\boldsymbol{0}\}$ then set $L := \{x | w_1a_1(x) + w_2a_2(x) = c\}$ is a line for all $c \in \mathbb{R}$. Henceforth, in the proofs of subsequent propositions, we will refer L as $w_1a_1 + w_2a_2 = c$ hiding the input variable, x, for ease of notation.*

Now we are ready to prove Proposition 2.5.8.

Let a_1, a_2 satisfy hard-sorting of $T_1 \cup T_0 \cup \{\mathbf{0}\}$ with respect to $T_1 \cup \{\mathbf{0}\}$. Then due to Remark 2.5.3, we have that a_1, a_2 satisfy hard-sorting of $T_1 \cup T_0$ with respect to T_1. We will show that any a_1, a_2 satisfying the above condition must satisfy the requirement of Proposition 2.5.8.

Let us partition the set of points $\mathbb{R}^2$ into four partitions $S_{0,0}, S_{+,0}, S_{0,+}$ and $S_{+,+}$ based on sign of $\big[a_1\big]_+$ and $\big[a_2\big]_+$. Then, we have to show that at least on element in T_1 lies in the partition $S_{0,0}$.

For sake of contradiction, assume that $T_1 \cap S_{0,0} = \varnothing$. Then, using pigeonhole principle, we have that at least one of $S_{+,0},\ S_{0,+}$ and $S_{+,+}$ must contain three points from the set T_1. Note that any three points in the set T_1 are not collinear. Moreover, the function $w_1\big[a_1\big]_+ + w_2\big[a_2\big]_+$ is affine in all three regions, $S_{+,0}, S_{0,+}$ and $S_{+,+}$ of $\mathbb{R}^2$ and is non-constant in view of Proposition 2.7.2. Hence, we cannot satisfy hard-sorting since those three points in T_1 will break the requirement in condition (2.3) for hard-sorting. Hence, we obtain a contradiction.

2.7.6 Proof of Proposition 2.7.2

First observe that if $a_1, a_2 : \mathbb{R}^2 \to \mathbb{R}$ satisfy hard-sorting of $T_1 \cup T_0 \cup \{\mathbf{0}\}$ with respect to $T_1 \cup \{\mathbf{0}\}$, then neither of them can be a constant function. In particular, it is straightforward to see that both of them cannot be constant. If only one of them is constant, then data needs to be linearly separable which is not the case for gadget data-points $T_1 \cup T_0 \cup \{\mathbf{0}\}$. Therefore, we will assume that both of them are affine functions with non-zero normal vectors.

Note that in view of Remark 2.5.4 and the fact that $w_1 a_1(x) + w_2 a_2(x) = c$ for all $x \in \mathbb{R}^2$, we may assume that magnitude of the normal to these lines is equal i.e. $\|\nabla a_1\| = \|\nabla a_2\| \neq 0$. For the sake of this proof, we extend the definition of hard-sorting to include the condition

$$w_1\big[a_1(x)\big]_+ + w_2\big[a_2(x)\big]_+ \begin{cases} = c & \text{for all } x \in T_1 \cup \{\mathbf{0}\}; \\ > c & \text{for all } x \in T_0, \end{cases}$$

along with condition (2.3). Due to this extended definition and in view of Remark 2.5.2, we just need to check for case $(w_1, w_2) = (1, 1)$ and $(w_1, w_2) = (1, -1)$. More specifically, $(w_1, w_2) = (-1, -1)$ yields a hard-sorting solution iff there exists a hard-sorting solution for $(w_1, w_2) = (1, 1)$. Equivalent argument can be made about the case $(w_1, w_2) = (-1, 1)$ and $(w_1, w_2) = (1, -1)$.

Then, we have two possible situations here: a_1, a_2 satisfy 1) $a_1(x) + a_2(x) = c, \forall\ x \in \mathbb{R}^2$ when normals point in opposite directions and 2) $a_1(x) - a_2(x) = c, \forall x \in \mathbb{R}^2$ when normals point in

same direction. We will consider both these cases separately and show that expression $w_1\big[a_1\big]_+ + w_2\big[a_2\big]_+$, for the choices of w_1, w_2 mentioned above, cannot hard-sort the data as required.

Case 1: Normals point in the opposite directions. Here $w_1 = w_2 = 1$ and we assume $a_1 + a_2 = c$. Suppose $c \geqslant 0$. Then it can be verified that

$$\big[a_1(x)\big]_+ + \big[a_2(x)\big]_+ = \begin{cases} c & \text{if } c \geqslant a_1(x) \geqslant 0 \\ a_1(x) & \text{if } a_1(x) \geqslant c \\ c - a_1(x) & \text{if } a_1(x) \leqslant 0. \end{cases}$$

By extended hard-sorting requirement, we need all points in $T_1 \cup \{\mathbf{0}\}$ should be contained in the set $\{x : a_1(x) \in [0, c]\}$ and all points in T_0 should not be in this set. Now observe that if $c = 0$, then the set $\{x : a_1(x) = 0\}$ is one dimensional, and therefore cannot contain all the points of $T_1 \cup \{\mathbf{0}\}$. Hence we must have $c > 0$ and all points in $T_1 \cup \{\mathbf{0}\}$ lie inside the region of two parallel lines $a_1(x) = 0$ and $a_1(x) = c$ as $\big[a_1(x)\big]_+ + \big[a_2(x)\big]_+$ evaluates to the constant c in this region. It can be seen that this separation of $T_1 \cup \{\mathbf{0}\}$ from T_0 is impossible to achieve by two parallel lines.

Similarly when $c < 0$, then it can be verified that

$$\big[a_1(x)\big]_+ + \big[a_2(x)\big]_+ = \begin{cases} 0 & \text{if } c \leqslant a_1(x) \leqslant 0 \\ a_1(x) & \text{if } a_1(x) \geqslant 0 \\ c - a_1(x) & \text{if } a_1(x) \leqslant c \end{cases}$$

Again, for extended hard-sorting, as in the previous case, we need all points in $T_1 \cup \{\mathbf{0}\}$ should be in set $\{x : a_1(x) \in [c, 0]\}$ and all points in T_0 should not be in this set which cannot be achieved.

Case 2: Normals point in the same direction. Then $a_1(x) - a_2(x) = c$. Suppose $c \geqslant 0$. Then it

can be verified that

$$\left[a_1(x)\right]_+ - \left[a_2(x)\right]_+ = \begin{cases} a_1(x) & \text{if } c \geqslant a_1(x) \geqslant 0 \\ c & \text{if } a_1(x) \geqslant c \\ 0 & \text{if } a_1(x) \leqslant 0 \end{cases}$$

If $c = 0$ then $\left[a_1(x)\right]_+ - \left[a_2(x)\right]_+ = 0$ for all $x \in \mathbb{R}^2$. So this cannot hard-sort data. Hence for hard-sorting we definitely need $c > 0$. Moreover, we need either 1) $T_1 \cup \{0\} \subset \{x : a_1(x) \leqslant 0\}$ and $T_0 \subset \{x : a_1(x) > 0\}$ or 2) $T_1 \cup \{0\} \subset \{x : a_1(x) \geqslant c\}$ and $T_0 \subset \{x : a_1(x) < c\}$. So essentially the points in $T_1 \cup T_0 \cup \{\mathbf{0}\}$ must be separable by a line. This is not possible.

Note that when $c < 0$, one can write $a_2 - a_1 = -c$ and write similar functional form for $\left[a_2\right]_+ - \left[a_1\right]_+$.

Since in both cases, we were unable to achieve hard-sorting $T_1 \cup T_0 \cup \{\mathbf{0}\}$ w.r.t. $T_1 \cup \{\mathbf{0}\}$, so we conclude the proof.

2.7.7 Proof of Lemma 2.5.9

Proposition 2.5.8 yields that any hard-sorting a_1, a_2 must satisfy $a_1(x) \leqslant 0, a_2(x) \leqslant 0$ for at least one $x \in T_1$.

Now, suppose sign of w_1, w_2 is different. Suppose $w_1 = 1, w_2 = -1$. Since a_1 and a_2 satisfy hard-sorting of gadget so we have $\left[a_1(x)\right]_+ - \left[a_2(x)\right]_+ = c, \forall\ x \in T_1$. Due to Proposition 2.5.8, we obtain $c = 0$. Then to fulfill hard-sorting condition, we need $\left[a_1(x)\right]_+ - \left[a_2(x)\right]_+ < 0\ \forall x \in T_0$. (The case for $w_1 = -1, w_2 = 1$ will have same proof with all a_2 exchanged by a_1 in next 3 lines.) This implies $a_2(x) > 0$ for all $x \in T_0$. However note that $T_1 \subset \text{conv}(T_0)$. So we get a contradiction to the assumption that sign of weights w_1, w_2 is different. Now note that if sign of w_1, w_2 is same then we cannot set $w_1 = w_2 = 1$ due to requirement (2.3) of hard-sorting. Hence we have that $w_1 = w_2 = -1$.

2.7.8 Proof of Corollary 2.5.11

The reduction is similar except the labels need to be changed from $\mathbb{R}$ to $\mathbb{R}^j$. Simply add $j-1$ zeros to original output labels. Now output of $j-1$ nodes is 0 for all data-points so these are redundant. In particular, for $k \in [j]$, every k-th node in the second layer is connected to 2 nodes in the first layer by distinct edges whose weights are parameterized by $w_{k,1}, w_{k,2}$ and bias weight $w_{k,0}$. We can set $w_{k,1} = w_{k,2} = -1$ and $w_{k,0} = 0$ for all $k \in [j] \backslash \{1\}$. This yields the output 0 at all nodes $k \in [j] \backslash \{1\}$, irrespective of the affine functions a_1, a_2 in the first layer. Now, first node satisfied to global optimality will yield solution $a_1, a_2, w_{1,1}, w_{1,2}, w_{1,0}$. By the reduction, we know that $-a_1, -a_2$ after ignoring last two co-ordinates yield solution to 2-hyperplane separability problem.

CHAPTER 3

STOCHASTIC FIRST-ORDER METHOD FOR CONVEX FUNCTION CONSTRAINED OPTIMIZATION

In the previous chapter, we saw convergence complexity for training neural networks. Henceforth, we will focus on algorithmic developments for function constrained optimization problems. In this chapter, our main focus will be on the development of efficient and simple algorithms for convex function constrained optimization. We will consider various settings of the convex function constrained problem, e.g., convex or strongly convex and Lipschitz smooth or nonsmooth objective and/or constraints which can be either stochastic or deterministic. We will present a novel algorithm that exhibits a unified convergence and reduces the impact of Lipschitz constants.

3.1 Convex Function Constrained Optimization Problem

In this paper, we study the following composite optimization problem with function constraints:

$$\begin{aligned} \min_{x\in X} \quad & \psi_0(x) := f_0(x) + \chi_0(x) \\ \text{s.t.} \quad & \psi_i(x) := f_i(x) + \chi_i(x) \leqslant 0, \quad i = 1, \ldots, m. \end{aligned} \tag{3.1}$$

Here, $X \subseteq \mathbb{R}^n$ is a convex compact set, $f_i : X \to \mathbb{R},\ i = 0, \ldots, m$ are continuous functions which are convex or strongly convex and $\chi_i : X \to \mathbb{R}, i = 0, \ldots, m$ are proper convex lower semicontinuous functions. Problem 3.1 covers different convex and strongly convex settings depending on the assumptions on f_i and χ_i, $i = 0, \ldots, m$.

In particular, we assume that f_i, $i = 0, \ldots, m$, are either smooth, nonsmooth or the sum of smooth and nonsmooth components. We also assume that χ_i, $i = 0, \ldots, m$, are "simple" functions in the sense that, for any given vector $v \in \mathbb{R}^n$ and non-negative weight vector $w \in \mathbb{R}^m$, a certain proximal operator associated with the function $\chi_0(x) + \sum_{i=1}^m w_i\chi_i(x) + \langle v, x\rangle$ can be computed

efficiently. For such problems, Lipschitz smoothness properties of χ_i's is of no consequence due to the simplicity of this proximal operator.

3.1.1 Algorithms for solving convex function constrained optimization

There exists a variety of literature on solving convex function constrained optimization problems (3.1). One research line focuses on primal methods without involving the Lagrange multipliers including the cooperative subgradient methods [90, 62] and level-set methods [63, 84, 66, 5, 65]. One possible limitation of these methods is the difficulty to directly achieve accelerated rate of convergence when the objective or constraint functions are smooth.

Constrained convex optimization problems can also be solved by reformulating them as saddle point problems which will then be solved by using primal-dual type algorithms (see [78, 46]). The main hurdle for existing primal-dual methods exists in that they require the projection of dual multipliers inside a ball whose diameter is usually unknown.

Other alternative approaches for constrained convex problems include the classical exact penalty, quadratic penalty and augmented Lagrangian methods [12, 55, 56, 113]. These approaches however require the solutions of penalty subproblems and hence are more complicated than primal and primal-dual methods.

Recently, research effort has also been directed to stochastic optimization problems with function constraints [62, 5]. In spite of many interesting findings, existing methods for solving these problems are still limited: a) many primal methods solve only stochastic problems with deterministic constraints [62], and the convergence for accelerated primal-dual methods [78, 46] has not been studied for stochastic function constrained problems; and b) a few algorithms for solving problems with expectation constraints require either a constraint evaluation step [62], or stochastic lower bounds on the optimal value [5], thus relying on a light-tail assumption for the stochastic noise and conservative sampling estimates based on Bernstein inequality. Some other algorithms require even more restrictive assumptions that the noise associated with stochastic constraints has to be bounded [117].

3.1.2 Unified algorithm for composite convex function constrained optimization

In this chapter, we attempt to address some of the aforementioned significant issues associated with both convex and nonconvex function constrained optimization.

Firstly, for solving convex function constrained problems, we present a novel primal-dual type method, referred to as the Constraint Extrapolation (ConEx) method. One distinctive feature of this method from existing primal-dual methods is that it utilizes linear approximations of the constraint functions to define the extrapolation (or acceleration/momentum) step. As a consequence, contrary to the well-known Nemirovski's mirror-prox method [78] and a primal-dual method recently developed by Hamedani and Aybat [46], ConEx does not require the projection of Lagrangian multipliers onto a (possibly unknown) bounded set. In addition, ConEx is a single-loop algorithm that does not involve any penalty subproblems. Due to the built-in acceleration step, this method can explore problem structures and hence achieve better rate of convergence than primal methods. In fact, we show that this method is a unified algorithm that achieves the optimal rate of convergence for solving different convex function constrained problems, including convex or strongly convex, and smooth or non-smooth problems with stochastic objective and/or stochastic constraints.

Table 3.1: Different convergence rates of the ConEx method for

	Strongly convex (3.1)		Convex (3.1)	
Cases	Smooth	Nonsmooth	Smooth	Nonsmooth
Deterministic	$O(1/\sqrt{\varepsilon})$	$O(1/\varepsilon)$	$O(1/\varepsilon)$	$O(1/\varepsilon^2)$
Semi-stochastic	$O(1/\varepsilon)$	$O(1/\varepsilon)$	$O(1/\varepsilon^2)$	$O(1/\varepsilon^2)$
Fully-stochastic	$O(1/\varepsilon^2)$	$O(1/\varepsilon^2)$	$O(1/\varepsilon^2)$	$O(1/\varepsilon^2)$

Table 3.1 provides a brief summary for the iteration complexity of the ConEx method for different problem settings such as strongly convex/convex, and smooth/nonsmooth objective and/or constraints. Deterministic means both objective and constraints are deterministic, semi-stochastic means objective is stochastic but constraints are deterministic, fully-stochastic means both objec-

tive and constraints are stochastic. For the strongly convex case, ConEx can obtain convergence to an ε-approximate solution (i.e., optimality gap and infeasibility are $O(\varepsilon)$) as well as convergence of the distance of the last iterate to the optimal solution. The complexity bounds provided in Table 3.1 for the strongly convex case hold for both types of convergence criteria. For semi-stochastic and fully-stochastic cases, we use the notion of expected convergence instead of exact convergence used in the deterministic case. It should be noted that in Table 3.1, we ignore the impact of various Lipschitz constants and/or stochastic noises for the sake of simplicity. In fact, the ConEx method achieves quite a few new complexity results by reducing the impact of these Lipschitz constants. This dependence is optimal for strongly convex case, in view of the lower bounds in [86] and is best-known for general convex problems. Moreover, to the best of our knowledge, it attains for the first time the optimal iteration and sampling complexity for solving general stochastic constrained problems without requiring the boundedness or light-tail assumptions on the stochastic subgradients (see Theorems 3.3.1 and 3.3.3 and discussions afterwards).

Even though ConEx is a primal-dual type method, we can show its convergence irrespective of the knowledge of the optimal Lagrange multipliers as it does not require the projection of multipliers onto the ball. In particular, convergence rates of the ConEx method for nonsmooth cases (either convex or strongly convex) in Table 3.1 holds irrespective of the knowledge of the optimal Lagrange multipliers. For smooth cases, if certain parameters of ConEx method are not big enough (compared to the norm of optimal Lagrange multipliers), then it converges at the rates for nonsmooth problems of the respective case. As one can see from Table 3.1, such a change would cause a suboptimal convergence rate in terms of ε only for the deterministic case, but complexity will be the same for both semi- and fully-stochastic cases.

It is worth mentioning that faster convergence rates for the smooth deterministic case can still be attained by incorporating certain line search procedures. ConEx method is arguably the first algorithm in the literature solving all different types of convex function constrained problems in an optimal and unified manner.

3.2 Notation and Terminologies

Throughout the paper, we use the following notations. Let

$$\begin{aligned}[m] &:= \{1, \ldots, m\}, \\ \psi(x) &:= [\psi_1(x), \ldots, \psi_m(x)]^T, \\ f(x) &:= [f_1(x), \ldots, f_m(x)]^T, \\ \chi(x) &:= [\chi_1(x), \ldots, \chi_m(x)]^T,\end{aligned} \tag{3.2}$$

and the constraints in (3.1) be expressed as $\psi(x) \leqslant \mathbf{0}$. Here bold $\mathbf{0}$ denotes the vector of elements 0. Size of the vector is left unspecified whenever it is clear from the context. $\|\cdot\|$ denotes a general norm and $\|\cdot\|_*$ denotes its dual norm defined as $\|z\|_* := \sup\{z^T x : \|x\| \leqslant 1\}$. From this definition, we obtain the $a^T b \leqslant \|a\| \|b\|_*$. Euclidean norm is denoted as $\|\cdot\|_2$ and standard inner product is denoted as $\langle \cdot, \cdot \rangle$. Let $\mathcal{B}^2(r) := \{x : \|x\|_2 \leqslant r\}$ be the Euclidean ball of radius r centered at origin. Nonnegative orthant of this ball is denoted as $\mathcal{B}^2_+(r)$. $\left[x\right]_+ := \max\{x, 0\}$ for any $x \in \mathbb{R}$. For any vector $x \in \mathbb{R}^k$, we define $\left[x\right]_+$ as element-wise application of the operator $\left[\,\cdot\,\right]_+$. The i-th element of vector x is denoted as x_i.

A function $r(\cdot)$ is λ-*Lipschitz smooth* if the gradient $\nabla r(x)$ is a λ-Lipschitz function, i.e. for some $\lambda \geqslant 0$

$$\|\nabla r(x) - \nabla r(y)\|_* \leqslant \lambda \|x - y\|, \;\; \forall x, y \in \operatorname{dom} r.$$

For a convex function r, an equivalent form of the above is:

$$0 \leqslant r(x) - r(y) - \langle \nabla r(y), x - y \rangle \leqslant \tfrac{\lambda}{2} \|x - y\|^2, \qquad \forall x, y \in \operatorname{dom} r.$$

In many cases, it is possible that a convex function r is a combination of Lipschitz smooth and nonsmooth functions. Let $\omega : X \to \mathbb{R}$ be continuously differentiable with L_ω Lipschitz gradient

and 1-strongly convex with respect to $\|\cdot\|$. We define the prox-function associated with $\omega(\cdot)$ as

$$W(y,x) := \omega(y) - \omega(x) - \langle \nabla\omega(x), y - x\rangle, \qquad \forall x, y \in X. \tag{3.3}$$

Based on the smoothness and strong convexity of $\omega(x)$, we have the following relation

$$W(y,x) \leqslant \tfrac{L_\omega}{2}\|x-y\|^2 \leqslant L_\omega W(x,y), \qquad \forall x, y \in X. \tag{3.4}$$

Moreover, we say that a function $r(\cdot)$ is β-strongly convex with respect to $W(\cdot,\cdot)$ if

$$r(x) \geqslant r(y) + \langle \nabla r(y), x - y\rangle + \beta W(x,y), \qquad \forall x, y \in X. \tag{3.5}$$

For any convex function h, we denote the subdifferential as ∂h which is defined as follows: at a point x in the relative interior of X, ∂h is comprised of all subgradients h' of h at x which are in the linear span of $X - X$. For a point $x \in X \setminus \operatorname{rint} X$, the set $\partial h(x)$ consists of all vectors h', if any, such that there exists $x_i \in \operatorname{rint} X$ and $h'_i \in \partial h(x_i), i = 1, 2, \ldots$, with $x = \lim\limits_{i\to\infty} x_i,\ h' = \lim\limits_{i\to\infty} h'_i$. With this definition, it is well-known that, if a convex function $h : X \to \mathbb{R}$ is Lipschitz continuous, with constant $\mathcal{M}$, with respect to a norm $\|\cdot\|$, then the set $\partial h(x)$ is nonempty for any $x \in X$ and

$$h' \in \partial h(x) \Rightarrow |\langle h', d\rangle| \leqslant \mathcal{M}\|d\|, \forall d \in \operatorname{lin}(X - X),$$

which also implies

$$h' \in \partial h(x) \Rightarrow \|h'\|_* \leqslant \mathcal{M},$$

where $\|\cdot\|_*$ is the dual norm. See [11] for more details.

3.3 Constraint Extrapolation Method

In this section, we present a novel constraint extrapolation (ConEx) method for solving problem (3.1). To motivate our proposed method, observe that the KKT point of (3.1) coincides with the

solution of the following saddle point problem:

$$\min_{x\in X}\max_{y\geqslant \mathbf{0}} \left\{\mathcal{L}(x,y) := \psi_0(x) + \textstyle\sum_{i=1}^m y^{(i)}\psi_i(x)\right\}. \tag{3.6}$$

In other words, (x^*, y^*) is a *saddle point* of the Lagrange function $\mathcal{L}(x, y)$ such that

$$\mathcal{L}(x^*, y) \leqslant \mathcal{L}(x^*, y^*) \leqslant \mathcal{L}(x, y^*), \tag{3.7}$$

for all $x \in X, y \geqslant \mathbf{0}$, whenever the optimal dual, y^*, exists. Throughout this chapter, we assume the existence of y^* satisfying (3.7). The following definition describes a widely used optimality measure for the convex problem (3.1).

Definition 3.3.1 *A point $\bar{x} \in X$ is called a (δ_o, δ_c)-optimal solution of problem* (3.1) *if*

$$\psi_0(\bar{x}) - \psi_0^* \leqslant \delta_0 \quad \textit{and} \quad \|\left[\psi(\bar{x})\right]_+\|_2 \leqslant \delta_c.$$

A stochastic (δ_o, δ_c)-approximately optimal solution satisfies

$$\mathbb{E}[\psi_0(\bar{x}) - \psi_0^*] \leqslant \delta_0 \quad \textit{and} \quad \mathbb{E}[\|\left[\psi(\bar{x})\right]_+\|_2] \leqslant \delta_c.$$

As mentioned earlier, for the convex composite case, we assume that $\chi_i, i = 0, \ldots, m$, are "simple" functions in the sense that, for any vector $v \in \mathbb{R}^n$ and nonnegative $w \in \mathbb{R}^m$, we can efficiently compute the following **prox** operator

$$\mathbf{prox}(w, v, \widetilde{x}, \eta) := \operatorname*{argmin}_{x\in X}\left\{\chi_0(x) + \textstyle\sum_{i=1}^m w_i\chi_i(x) + \langle v, x\rangle + \eta W(x, \widetilde{x})\right\}. \tag{3.8}$$

ConEx is a single-loop primal-dual type method for function constrained optimization. It evolves from the primal-dual methods for solving bilinear saddle point point problems (e.g., [22, 24, 61, 58, 54]). Recently Hamedani and Aybat [46] show that these methods can also handle more general function coupling term. However, as discussed earlier, existing primal-dual methods [78,

46] for general saddle point problems, when applied to function constrained problems, require the projection of dual multipliers onto a possibly unknown bounded set in order to ensure the boundedness of the multipliers, as well as the proper selection of stepsizes. One distinctive feature of ConEx is to use value of linearized constraint functions in place of exact function values when defining the operator of the saddle point problem and the extrapolation/momentum step. With this modification, we show that the ConEx method still converges even though the feasible set of y in problem (3.6) is unbounded.

In addition, we show that the ConEx is a unified algorithm for solving function constrained optimization problems in the following sense. First, we establish explicit rate of convergence for the ConEx method for solving function constrained stochastic optimization problems where either the objective and/or constraints are given in the form of expectation. Second, we consider the composite constrained optimization problem in which objective function f_0 and/or constraints $f_i, i = 1, \ldots, m$ can be nonsmooth. Third, we consider the two cases of convex or strongly convex objective, f_0. For strongly convex objective, we also establish the convergence rate of the distance between last iterate to the optimal solution x^*.

Before proceeding to the algorithm, we introduce the problem setup in more details. First, we assume that f_0 satisfies the following Lipschitz smoothness and nonsmoothness condition:

$$f_0(x_1) - f_0(x_2) - \langle f_0'(x_2), x_1 - x_2 \rangle \leqslant \tfrac{L_0}{2}\|x_1 - x_2\|^2 + H_0\|x_1 - x_2\| \tag{3.9}$$

for all $x_1, x_2 \in X$ and for all $f_0'(x_2) \in \partial f_0(x_2)$. For constraints, we make a similar assumption as in (3.9). Moreover, we make an additional assumption that the constraint functions are Lipschitz continuous. In particular, we have

$$f_i(x_1) - f_i(x_2) - \langle f_i'(x_2), x_1 - x_2 \rangle \leqslant \tfrac{L_i}{2}\|x_1 - x_2\|^2 + H_i\|x_1 - x_2\|, \tag{3.10}$$

for all $x_1, x_2 \in X$ and for all $f_i'(x_2) \in \partial f_i(x_2),\ i = 1, \ldots, m$, and

$$\begin{aligned} f_i(x_1) - f_i(x_2) &\leqslant M_{f,i}\|x_1 - x_2\|, \quad \forall x_1, x_2 \in X, i = 1, \ldots, m, \\ \chi_i(x_1) - \chi_i(x_2) &\leqslant M_{\chi,i}\|x_1 - x_2\|, \quad \forall x_1, x_2 \in X, i = 1, \ldots, m. \end{aligned} \tag{3.11}$$

Note that the Lipschitz-continuity assumption in (3.11) is common in the literature when $f_i, i \in [m]$, are nonsmooth functions. If $f_i, i \in [m]$, are Lipschitz smooth then their gradients are bounded due to the compactness of X. Hence (3.11) is not a strong assumption for the given setting. Also note that due to definition of subgradient for convex function defined in Section 3.2, we have $\|f_i'(\cdot)\|_* \leqslant M_{f,i}$ which implies $|f_i'(x_2)^T(x_1 - x_2)| \leqslant \|f_i'(x_2)\|_*\|x_1 - x_2\| \leqslant M_{f,i}\|x_1 - x_2\|$.

Using this relation and noting relations (3.10) and (3.11), we have the following four relations:

$$\begin{aligned} \|f(x_1) - f(x_2)\|_2 &\leqslant M_f\|x_1 - x_2\|, \\ \|\chi(x_1) - \chi(x_2)\|_2 &\leqslant M_\chi\|x_1 - x_2\|, \\ \|f(x_1) - f(x_2) - f'(x_2)^T(x_1 - x_2)\|_2 &\leqslant \tfrac{L_f}{2}\|x_1 - x_2\|^2 + H_f\|x_1 - x_2\|, \\ \|f'(x_2)^T(x_1 - x_2)\|_2 &\leqslant M_f\|x_1 - x_2\|, \end{aligned} \tag{3.12}$$

for all $x_1, x_2 \in X$. Here $f'(\cdot) := [f_1'(\cdot), \ldots, f_m'(\cdot)] \in \mathbb{R}^{n\times m}$ and constants M_f, M_χ, H_f and L_f are defined as

$$\begin{aligned} M_f &:= (\textstyle\sum_{i=1}^m M_{f,i}^2)^{1/2}, \quad M_\chi := (\sum_{i=1}^m M_{\chi,i}^2)^{1/2}, \\ H_f &:= (\textstyle\sum_{i=1}^m H_i^2)^{1/2}, \qquad L_f := (\sum_{i=1}^m L_i^2)^{1/2}. \end{aligned} \tag{3.13}$$

We denote $\alpha = (\alpha_1, \ldots, \alpha_m)^T$ as the vector of moduli of strong convexity for $\chi_i, i \in [m]$, and α_0 as the modulus of strong convexity for χ_0. We say that problem (3.1) is a convex composite smooth (also referred to as composite smooth) function constrained minimization problem if (3.10) is satisfied with $H_i = 0$ for all $i = 1, \ldots, m$ and (3.9) is satisfied with $H_0 = 0$. Otherwise, (3.1) is a nonsmooth problem. To be succinct, problem (3.1) is composite smooth if $H_f = H_0 = 0$, otherwise it is a nonsmooth problem.

We assume that we can access the first-order information of functions f_0, f_i and zeroth-order information of function f_i using a stochastic oracle (SO). In particular, given $x \in X$, SO outputs $G_0(x,\xi), G_i(x,\xi)$, and $F(x,\xi)$ such that

$$\begin{aligned}
\mathbb{E}[G_0(x,\xi)] &= f_0'(x), \\
\mathbb{E}[G_i(x,\xi)] &= f_i'(x), \qquad i = 1,\dots,m, \\
\mathbb{E}[F(x,\xi)] &= f(x), \\
\mathbb{E}\big[\|G_0(x,\xi) - f_0'(x)\|_*^2\big] &\leqslant \sigma_0^2, \\
\mathbb{E}\big[\|G_i(x,\xi) - f_i'(x)\|_*^2\big] &\leqslant \sigma_i^2, \qquad i = 1,\dots,m, \\
\mathbb{E}[\|F(x,\xi) - f(x)\|_2^2] &\leqslant \sigma_f^2,
\end{aligned} \tag{3.14}$$

where ξ is a random variable which models the source of uncertainty and is independent of the search point x. Note that the last relation of (3.14) is satisfied if we have individual stochastic oracles $F_i(x,\xi)$ such that $\mathbb{E}[(F_i(x,\xi) - f_i(x))^2] \leqslant \sigma_{f,i}^2$. In particular, we can set $\sigma_f^2 = \sum_{i=1}^m \sigma_{f,i}^2$. We call $G_i, i = 0,\dots,m$, as stochastic subgradients of functions $f_i, i = 0,\dots,m$ at point x, respectively. We use stochastic subgradients $G_i(x_t,\xi_t),\ i = 0,\dots,m$, in the t-th iteration of the ConEx method where ξ_t is a realization of random variable ξ which is independent of the search point x_t.

We denote $\ell_f^{t-1}(x_t)$ a linear approximation of $f(\cdot)$ at point x_t with

$$\ell_f^{t-1}(x_t) := f(x_{t-1}) + f'(x_{t-1})^T(x_t - x_{t-1}),$$

where $f'(x_{t-1}) = [f_1'(x_{t-1}),\dots,f_m'(x_{t-1})]$ as defined earlier. For ease of notation, we denote $\ell_f^{t-1}(x_t)$ as $\ell_f(x_t)$. We can do this, since for all t, we approximate $f(x_t)$ with linear function approximation taken at x_{t-1}. We use a stochastic version of ℓ_f in our algorithm, which is denoted as ℓ_F. In particular, we have

$$\ell_F(x_t) := F(x_{t-1},\bar{\xi}_{t-1}) + \mathbf{G}(x_{t-1},\bar{\xi}_{t-1})^T(x_t - x_{t-1}),$$

where $\mathbf{G}(x_{t-1}, \bar{\xi}_{t-1}) := [G_1(x_{t-1}, \bar{\xi}_{t-1}), \ldots, G_m(x_{t-1}, \bar{\xi}_{t-1})] \in \mathbb{R}^{n\times m}$. Here, we used $\bar{\xi}_t$ as an independent (of ξ_t) realization of random variable ξ. In other words, $G_i(x_t, \bar{\xi}_t)$ and $G_i(x_t, \xi_t)$ are conditionally independent estimates of $f_i'(x_t)$ for $i = 1, \ldots, m$ under the condition that x_t is fixed. As we show later, independent samples of ξ are required to show that $\ell_F(x_t)$ is an unbiased estimator of $\ell_f(x_t)$.

We are now ready to formally describe the constraint extrapolation method (see Algorithm 1). As mentioned earlier, the $\ell_F(x_t)$ term in Line 3 of Algorithm 1 can be shown to be an unbiased

Algorithm 1 Constraint **Ex**trapolation (ConEx) Method

Input: $(x_0, y_0), \{\gamma_t, \tau_t, \eta_t, \theta_t\}_{t\geqslant 0}, T$.

1: $(x_{-1}, y_{-1}) \leftarrow (x_0, y_0)$, $F(x_{-1}) \leftarrow F(x_0, \bar{\xi}_0)$ and $\ell_F(x_{-1}) \leftarrow \ell_F(x_0)$
2: **for** $t = 0, \ldots, T-1$ **do**
3: $\quad s_t \leftarrow (1+\theta_t)[\chi(x_t) + \ell_F(x_t)] - \theta_t[\chi(x_{t-1}) + \ell_F(x_{t-1})]$.
4: $\quad y_{t+1} \leftarrow \left[y_t + \frac{1}{\tau_t} s_t\right]_+$.
5: $\quad x_{t+1} \leftarrow \mathbf{prox}\big(y_{t+1}, G_0(x_t, \xi_t) + \sum_{i\in[m]} G_i(x_t, \xi_t) y_{t+1}^{(i)}, x_t, \eta_t\big)$.
6: **end for**
7: **return** $\bar{x}_T = \left(\sum_{t=0}^{T-1} \gamma_t\right)^{-1} \sum_{t=0}^{T-1} \gamma_t x_{t+1}$.

estimator of $\ell_f(x_t)$. Moreover, the term $\chi(x_t) + \ell_f(x_t)$ is an approximation to $\chi(x_t) + f(x_t) = \psi(x_t)$. Essentially, Line 3 represents a stochastic approximation for the term $\psi(x_t) + \theta_t(\psi(x_t) - \psi(x_{t-1}))$ which is an extrapolation of the constraints, hence justifying the name of the algorithm. Line 4 is the standard **prox** operator of the form $\operatorname{argmin}_{y\geqslant \mathbf{0}} \langle -s_t, y\rangle + \frac{\tau_t}{2}\|y - y_t\|_2^2$. Line 5 also uses a prox operator defined in (3.8) which uses Bregman divergence W instead of standard Euclidean norm. The final output of the algorithm in Line 7 is the weighted average of all primal iterates generated. If we choose $\sigma_f = \sigma_0 = \sigma_i = 0$ for $i = 1, \ldots, m$ then we recover the deterministic gradients and function evaluation. Henceforth, we assume general non-negative values for such σ's and provide a combined analysis for these settings. Later, we substitute appropriate values of σ's to finish the analysis for the following three different cases.

a) Deterministic setting where both the objective and constraints are deterministic. Here $\sigma_0 = \sigma_i = \sigma_\psi = 0$ for all $i \in [m]$.

b) Semi-stochastic setting where the constraints are deterministic but the objective is stochastic. Here, $\sigma_\psi = \sigma_i = 0$ for all $i \in [m]$. However, $\sigma_0 \geqslant 0$ can take arbitrary values.

c) Fully-stochastic setting where both function and gradient evaluations are stochastic. Here, all $\sigma_\psi, \sigma_0, \sigma_i \geqslant 0$ can take arbitrary values.

Below, we specify a stepsize policy and state the convergence properties of Algorithm 1 for solving problem (3.1) in the strongly convex setting. The proof of this result is involved and will be deferred to Section 3.4.

Theorem 3.3.1 *Suppose* (3.9), (3.10), (3.11) *and* (3.14) *are satisfied. Let* $B \geqslant 1$ *be a constant,* $t_0 := \frac{4(L_0+BL_f)}{\alpha_0} + 2$, $\mathcal{M} := \max\{2M_f, M_\chi + M_f\}$, *and* $\sigma_{X,f} := (\sigma_f^2 + D_X^2\|\sigma\|_2^2)^{1/2}$. *Set* $y_0 = \mathbf{0}$ *and* $\{\gamma_t, \theta_t, \eta_t, \tau_t\}$ *in Algorithm 1 according to the following:*

$$\begin{aligned} &\gamma_t = t + t_0 + 2, && \eta_t = \tfrac{\alpha_0(t+t_0+1)}{2}, \\ &\tau_t = \tfrac{1}{t+1}\max\{\tfrac{32\mathcal{M}^2}{\alpha_0}, \tfrac{384\|\sigma\|_2^2 T}{\alpha_0}, \tfrac{\sigma_{X,f}T^{3/2}}{B(t_0+2)^{1/2}}\}, && \theta_t = \tfrac{t+t_0+1}{t+t_0+2}. \end{aligned} \tag{3.15}$$

Then for $T \geqslant 1$, *we have*

$$\mathbb{E}[\psi_0(\bar{x}_T) - \psi_0(x^*)] \leqslant \tfrac{\alpha_0(t_0+1)(t_0+2)D_X^2}{T^2} + \tfrac{12B\sigma_{X,f}(t_0+1)(t_0+2)^{1/2}}{T^{3/2}} + \tfrac{16(\zeta^2+H_0^2)}{\alpha_0 T} + \tfrac{8B(t_0+2)^{1/2}\sigma_{X,f}}{T^{1/2}}. \tag{3.16}$$

and

$$\begin{aligned} \mathbb{E}\Big\|\big[\psi(\bar{x}_T)\big]_+\Big\|_2 \leqslant\ & \tfrac{192(t_0+2)(\|y^*\|_2+1)^2\mathcal{M}^2}{\alpha_0 T^2} + \tfrac{\alpha_0(t_0+1)(t_0+2)D_X^2}{T^2} + \tfrac{13B\sigma_{X,f}(t_0+1)(t_0+2)^{1/2}}{T^{3/2}} \\ &+ \tfrac{16(\zeta^2+\mathcal{H}_*^2+144(t_0+2)(\|y^*\|_2+1)^2\|\sigma\|_2^2)}{\alpha_0 T} \\ &+ \Big\{\tfrac{6(t_0+2)^{1/2}(\|y^*\|_2+1)^2\sigma_{X,f}}{B} + \tfrac{26B(t_0+2)^{1/2}\sigma_{X,f}}{3}\Big\}\tfrac{1}{T^{1/2}}, \end{aligned} \tag{3.17}$$

where

$$\mathcal{H}_* := H_0 + (\|y^*\|_2 + 1)H_f + \tfrac{L_f D_X[\|y^*\|_2+1-B]_+}{2},$$
$$\zeta := 2e\Big\{\big[\sigma_0^2 + 12(t_0+3)\|\sigma\|_2^2\|y^*\|_2^2 + 96(t_0+2)B^2\|\sigma\|_2^2 + \tfrac{\mathcal{H}_*^2}{2} + \tfrac{3\alpha_0 B\sigma_{X,f}(t_0+2)^{3/2}}{2}\big]\Big\}^{1/2}.$$

Moreover, we obtain the last iterate convergence

$$\begin{aligned}\mathbb{E}[W(x^*, X_T)] \leqslant\ & \tfrac{192(t_0+2)(\|y^*\|_2+1)^2\mathcal{M}^2}{\alpha_0^2 T^2} + \tfrac{(t_0+1)(t_0+2)D_X^2}{T^2} + \tfrac{12B\sigma_{X,f}(t_0+1)(t_0+2)^{1/2}}{\alpha_0 T^{3/2}} \\ & + \tfrac{16(\zeta^2+\mathcal{H}_*^2+144(t_0+2)(\|y^*\|_2+1)^2\|\sigma\|_2^2)}{\alpha_0^2 T} \\ & + \tfrac{(t_0+2)^{1/2}\|y^*\|_2^2\sigma_{X,f}}{B\alpha_0}\tfrac{1}{T^{1/2}} + \tfrac{8B(t_0+2)^{1/2}\sigma_{X,f}}{\alpha_0 T^{1/2}}.\end{aligned} \tag{3.18}$$

An immediate corollary of the above theorem is the following:

Corollary 3.3.2 *We obtain an $(\varepsilon, \varepsilon)$-optimal solution of problem* (3.1) *in T_ε iterations, where*

$$\begin{aligned}T_\varepsilon = \max\Big\{&\Big(\tfrac{5\alpha_0(t_0+2)(t_0+1)D_X^2}{\varepsilon} + \tfrac{960(t_0+2)(\|y^*\|_2+1)^2\mathcal{M}^2}{\alpha_0\varepsilon}\Big)^{1/2}, \big(\tfrac{65B\sigma_{X,f}(t_0+2)^{3/2}}{\varepsilon}\big)^{2/3}, \\ &\tfrac{80(\zeta^2+\mathcal{H}_*^2+144(t_0+2)(\|y^*\|_2+1)^2\|\sigma\|_2^2)}{\alpha_0\varepsilon}, \big(\tfrac{30(\|y^*\|_2+1)^2\sigma_{X,f}}{B}\big)\tfrac{t_0+2}{\varepsilon^2}, \big(\tfrac{130B\sigma_{X,f}}{3}\big)^2\tfrac{t_0+2}{\varepsilon^2}\Big\}.\end{aligned} \tag{3.19}$$

Moreover, we obtain $\mathbb{E}[W(x^*, x_T)] \leqslant \varepsilon$ in at most

$$\begin{aligned}\max\Big\{&\Big(\tfrac{5(t_0+2)(t_0+1)D_X^2}{\varepsilon} + \tfrac{960(t_0+2)(\|y^*\|_2+1)^2\mathcal{M}^2}{\alpha_0^2\varepsilon}\Big)^{1/2}, \big(\tfrac{60B\sigma_{X,f}(t_0+2)^{3/2}}{\alpha_0\varepsilon}\big)^{2/3}, \\ &\tfrac{80(\zeta^2+\mathcal{H}_*^2+144(t_0+2)(\|y^*\|_2+1)^2\|\sigma\|_2^2)}{\alpha_0^2\varepsilon}, \big(\tfrac{5\|y^*\|_2^2\sigma_{X,f}}{B\alpha_0}\big)^2\tfrac{t_0+2}{\varepsilon^2}, \big(\tfrac{40B\sigma_{X,f}}{\alpha_0}\big)^2\tfrac{t_0+2}{\varepsilon^2}\Big\}\end{aligned} \tag{3.20}$$

iterations.

Proof. Using (3.17) and (3.19), we have $\mathbb{E}\Big\|\big[\psi(\bar{x}_T)\big]_+\Big\|_2 \leqslant \frac{\varepsilon}{5} + \frac{\varepsilon}{5} + \frac{\varepsilon}{5} + \frac{\varepsilon}{5} + \frac{\varepsilon}{5} = \varepsilon$. Similarly, using (3.16) and (3.19), it is easy to observe that $\mathbb{E}[\psi_0(\bar{x}_T) - \psi_0(x^*)] \leqslant \varepsilon$. Using (3.18) and (3.20), we have $\mathbb{E}[W(x^*, x_T)] \leqslant \frac{\varepsilon}{5} + \frac{\varepsilon}{5} + \frac{\varepsilon}{5} + \frac{\varepsilon}{5} + \frac{\varepsilon}{5} = \varepsilon$. Hence we conclude the proof. □

Theorem 3.3.1 and Corollary 3.3.2 provide unified iteration complexity bounds for solving

strongly convex function constrained optimization problems. These results will also be used later for solving subproblems arising from the proximal point method for nonconvex problems in Section 4.2. Below we derive from (3.19) the convergence rate of Algorithm 1 for both nonsmooth problems, i.e., either H_f or H_0 is strictly positive, and (composite) smooth problems, i.e., $H_f = 0, H_0 = 0$.

Let us start with nonsmooth problems for which (3.9) is satisfied with $H_0 > 0$ or (3.10) is satisfied with $H_i > 0$ for at least one $i \in [m]$. In this case, we have

$$\mathcal{H}_* = (\|y^*\|_2 + 1)H_f + H_0 + \tfrac{L_f D_X[\|y^*\|_2 + 1 - B]_+}{2} > 0$$

irrespective of the value of B. Then, using (3.19), we obtain the iteration complexity of

$$O\Big(\tfrac{1}{\sqrt{\varepsilon}}\big(\tfrac{(L_0+BL_f)D_X}{\sqrt{\alpha_0}} + \tfrac{\sqrt{L_0+BL_f}B\mathcal{M}}{\alpha_0}\big) + \tfrac{\mathcal{H}_*^2}{\alpha_0\varepsilon}\Big)$$

for the deterministic case. For the semi-stochastic case, the iteration complexity becomes

$$O\Big(\tfrac{1}{\sqrt{\varepsilon}}\big(\tfrac{(L_0+BL_f)D_X}{\sqrt{\alpha_0}} + \tfrac{\sqrt{L_0+BL_f}B\mathcal{M}}{\alpha_0}\big) + \tfrac{(\mathcal{H}_*^2+\sigma_0^2)}{\alpha_0\varepsilon}\Big).$$

Similarly, for the fully-stochastic case, the iteration complexity is given by

$$O\Big(\tfrac{1}{\sqrt{\varepsilon}}\big(\tfrac{(L_0+BL_f)D_X}{\sqrt{\alpha_0}} + \tfrac{\sqrt{L_0+BL_f}B\mathcal{M}}{\alpha_0}\big) + \tfrac{(\mathcal{H}_*^2+\zeta^2)}{\alpha_0\varepsilon} + \tfrac{1}{\varepsilon^2}\big\{\tfrac{B^2(L_0+BL_f)(\sigma_0^2+D_X^2\|\sigma\|_2^2)}{\alpha_0}\big\}\Big).$$

Observe that, due to the built-in acceleration scheme of the ConEx method, the Lipschitz constant L_0 will barely impact the convergence since it appears only in the $O(1/\sqrt{\varepsilon})$ term. Similarly, the impact of the Lipschitz constant L_f will be minimized for a large enough B (i.e., $B \geqslant \|y^*\|_2 + 1$). To the best of our knowledge, these complexity results with separate impact of Lipschitz constants appear to be new for function constrained optimization. Moreover, the iteration (and sample) complexity for the fully-stochastic case, i.e., general stochastic constrained problems requiring

only bounded second moments on nosies, has not been obtained before in the literature.

Now let us consider smooth problems for which (3.9) and (3.10) are satisfied with $H_0 = 0$ and $H_i = 0$ for all $i = 1, \ldots, m$, respectively. We distinguish two different scenarios depending on whether $B \geqslant \|y^*\|_2 + 1$. First, if $B \geqslant \|y^*\|_2 + 1$, then $\mathcal{H}_* = H_0 + H_f(\|y^*\|_2 + 1) + L_f D_X[\|y^*\|_2 + 1 - B]_+/2 = 0$ and the iteration complexity in (3.19) can be simplified as follows. For the deterministic case, the iteration complexity in (3.19) reduces to

$$O\Big(\tfrac{1}{\sqrt{\varepsilon}}\big(\tfrac{(L_0+BL_f)D_X}{\sqrt{\alpha_0}} + \tfrac{\sqrt{L_0+BL_f}B\mathcal{M}}{\alpha_0}\big)\Big). \tag{3.21}$$

Moreover, the complexity bounds for the semi- and fully-stochastic cases are given by

$$O\Big(\tfrac{1}{\sqrt{\varepsilon}}\big(\tfrac{(L_0+BL_f)D_X}{\sqrt{\alpha_0}} + \tfrac{\sqrt{L_0+BL_f}B\mathcal{M}}{\alpha_0}\big) + \tfrac{\sigma_0^2}{\alpha_0\varepsilon}\Big), \tag{3.22}$$

$$O\Big(\tfrac{1}{\sqrt{\varepsilon}}\big(\tfrac{(L_0+BL_f)D_X}{\sqrt{\alpha_0}} + \tfrac{\sqrt{L_0+BL_f}B\mathcal{M}}{\alpha_0}\big) + \tfrac{\zeta^2}{\alpha_0\varepsilon} + \tfrac{1}{\varepsilon^2}\big\{\tfrac{B^2(L_0+BL_f)(\sigma_0^2+D_X^2\|\sigma\|_2^2)}{\alpha_0}\big\}\Big), \tag{3.23}$$

respectively, where $\zeta^2 = O(\sigma_0^2 + B^2(L_0 + BL_f)\|\sigma\|_2^2/\alpha_0)$. It is worth noting that a similar bound to 3.21 has been obtained in [46] with a slightly different termination criterion[1]. On the other hand, the complexity bounds in 3.22 and 3.23 for the semi-stochastic and fully-stochastic cases seem to be new in the literature.

Second, if $B < \|y^*\|_2 + 1$ for the smooth case, then $\mathcal{H}_* > 0$ and the ConEx method converges at the rate of nonsmooth problems in all these three settings described above. Hence, the ConEx method still converges albeit at a slower rate without knowing exact bound on $\|y^*\|_2$. On the other hand, existing primal-dual methods require correct estimation of $\|y^*\|_2$ in order to define the projection operator and properly select stepsize. Observe that one can possibly perform a line search for right value of B when specifying τ_t in the ConEx method in order to obtain a faster convergence rate, especially for the deterministic and semi-stochastic cases where the constraint

[1]The infeasibility in [46] is measured by $y^*\big[\psi(\bar{x}_T)\big]_+$, and hence may vanish for constraints with $y_i^* = 0$.

violations $\|\big[\psi(\cdot)\big]_+\|_2$ can be measured precisely.

It is worth mentioning that for the complexity results discussed above, we do not require the constraints $\psi_i,\ i = 1, \ldots, m$, to be strongly convex. From (3.15), we can see that $\alpha_0 > 0$ is enough to ensure the selection of stepsize policy which yields accelerated convergence rates. In particular, if $\alpha_i = 0$ for all $i \in [m]$ (implying ψ_i's are merely convex functions) then η_t in relation (3.34) is required to satisfy the following more stringent relation: $\gamma_t\eta_t \leqslant \gamma_{t-1}(\eta_{t-1} + \alpha_0)$. Note that our stepsize policy already satisfies this relation. Hence Algorithm 1 exhibits accelerated convergence rates even if the constraints are merely convex.

Now we provide another theorem which states the stepsize policy and the resulting convergence properties of the ConEx method for solving problem (3.1) without any strong convexity assumptions. The proof of this result can be found in Section 3.4.

Theorem 3.3.3 *Suppose* (3.9)*,* (3.10)*,* (3.11) *and* (3.14) *are satisfied. Let* $B \geqslant 1$ *be a given constant,* $\mathcal{M}$*,* $\sigma_{X,f}$ *and* $\mathcal{H}_*$ *be defined as in Theorem 3.3.1. Set* $y_0 = \mathbf{0}$ *and* $\{\gamma_t, \theta_t, \eta_t, \tau_t\}$ *in Algorithm 1 according to the following:*

$$\begin{aligned} &\gamma_t = 1, \quad \eta_t = L_0 + BL_f + \eta, \\ &\theta_t = 1, \quad \tau_t = \tau, \end{aligned} \tag{3.24}$$

where

$$\begin{aligned} \eta &:= \max\Big\{\tfrac{\sqrt{2T[\mathcal{H}_*^2+\sigma_0^2+48B^2\|\sigma\|_2^2]}}{D_X}, \tfrac{6B\max\{\mathcal{M},4\|\sigma\|_2\}}{D_X}\Big\}, \\ \tau &:= \max\Big\{\tfrac{\sqrt{96T}\sigma_{X,f}}{B}, \tfrac{2D_X\max\{\mathcal{M},4\|\sigma\|_2\}}{B}\Big\}. \end{aligned}$$

Then, we have

$$\mathbb{E}[\psi_0(\bar{x}_T) - \psi_0(x^*)] \leqslant \tfrac{(L_0+BL_f)D_X^2+\max\{6\mathcal{M},24\|\sigma\|_2\}BD_X}{T} + \tfrac{1}{\sqrt{T}}\Big\{\tfrac{\sqrt{2}(\zeta^2+H_0^2)D_X}{\sqrt{\mathcal{H}_*^2+\sigma_0^2+48B^2\|\sigma\|_2^2}} + \tfrac{\sqrt{3}B\sigma_{X,f}}{\sqrt{2}}\Big\} \tag{3.25}$$

and

$$\mathbb{E}[\|\big[\psi(\bar{x}_T)\big]_+\|_2] \leqslant \frac{(L_0+BL_f)D_X^2+\max\{6\mathcal{M},24\|\sigma\|_2\}D_X\left(B+\frac{(\|y^*\|_2+1)^2}{B}\right)}{T} + \frac{1}{\sqrt{T}}\Big\{\big[\frac{12\sqrt{6}(\|y^*\|_2+1)^2}{B}+\frac{13B}{4\sqrt{6}}\big]\sigma_{X,f} + \sqrt{2}D_X\big[\sqrt{\mathcal{H}_*^2+B^2\sigma_0^2+48\|\sigma\|_2^2}+\frac{\zeta^2+\mathcal{H}_*^2}{\sqrt{\mathcal{H}_*^2+\sigma_0^2+48B^2\|\sigma\|_2^2}}\big]\Big\}, \tag{3.26}$$

where

$$\zeta := 2e\big\{\sigma_0^2+\|\sigma\|_2^2(14\|y^*\|_2^2+123B^2)+2\sqrt{3}\|\sigma\|_2(2B\mathcal{H}_*+B\sigma_0)\big\}^{1/2}.$$

As a consequence, the number of iterations performed by Algorithm 1 to find an $(\varepsilon,\varepsilon)$*-optimal solution of problem* (3.1) *can be bounded by*

$$\max\Big\{\frac{3(L_0+BL_f)D_X^2+\max\{36\mathcal{M},144\|\sigma\|_2\}(\|y^*\|_2+1)D_X}{\varepsilon}, \big[\frac{36\sqrt{6}(\|y^*\|_2+1)^2}{B}+\frac{13\sqrt{3}B}{4\sqrt{2}}\big]^2\frac{\sigma_{X,f}^2}{\varepsilon^2}, \frac{18}{\varepsilon^2}\big[D_X\sqrt{\mathcal{H}_*^2+\sigma_0^2+48B^2\|\sigma\|_2^2}+\frac{D_X(\zeta^2+\mathcal{H}_*^2)}{\sqrt{\mathcal{H}_*^2+\sigma_0^2+48B^2\|\sigma\|_2^2}}\big]^2\Big\}. \tag{3.27}$$

Theorem 3.3.3 provides unified iteration complexity bounds for solving convex function constrained optimization problems. Below we derive from (3.27) the convergence rate of Algorithm 1 for solving both nonsmooth problems, i.e., either H_f or H_0 is strictly positive, and (composite) smooth problems, i.e., $H_f=0, H_0=0$.

Let us start with the more general nonsmooth problems. Since $H_i>0$ for some $i=0,\ldots,m$, we have $\mathcal{H}_*>0$. Then, the complexity bound in (3.27) for the deterministic, semi-stochastic and fully-stochastic cases, respectively, will reduce to

$$O\big(\frac{L_0+BD_X(L_fD_X+\mathcal{M})}{\varepsilon}+\frac{D_X^2\mathcal{H}_*^2}{\varepsilon^2}\big),$$
$$O\big(\frac{L_0+BD_X(L_fD_X+\mathcal{M})}{\varepsilon}+\frac{D_X^2(\mathcal{H}_*^2+\sigma_0^2)}{\varepsilon^2}\big),$$

and

$$O\Big(\tfrac{L_0+BD_X(L_fD_X+\mathcal{M})}{\varepsilon}+\tfrac{B^2(\sigma_f^2+D_X^2\|\sigma\|_2^2)+D_X^2(\sigma_0^2+\mathcal{H}_*^2)}{\varepsilon^2}\Big). \tag{3.28}$$

Similarly to the strongly convex case, the separate impact of the Lipschitz constants (L_0 and L_f) on these complexity bounds have not been obtained before. Moreover, the iteration (and sampling) complexity for the fully-stochastic case, i.e., general stochastic constrained problems requiring only bounded second moments on nosies, appears to be new in the literature.

Now let us consider smooth problems for which $H_f = H_0 = 0$. We distinguish two different scenarios depending on whether $B \geqslant \|y^*\|_2 + 1$. First, if $B \geqslant \|y^*\|_2 + 1$, then $\mathcal{H}_* = 0$ and the complexity bound in (3.27) for the deterministic, semi-stochastic and fully-stochastic cases, respectively, will reduce to

$$O\big(\tfrac{L_0+BD_X(L_fD_X+\mathcal{M})}{\varepsilon}\big), \tag{3.29}$$

$$O\big(\tfrac{L_0+BD_X(L_fD_X+\mathcal{M})}{\varepsilon}+\tfrac{\sigma_0^2D_X^2}{\varepsilon^2}\big), \tag{3.30}$$

and

$$O\Big(\tfrac{L_0+BD_X(L_fD_X+\mathcal{M})}{\varepsilon}+\tfrac{B^2(\sigma_f^2+D_X^2\|\sigma\|_2^2)+D_X^2\sigma_0^2}{\varepsilon^2}\Big), \tag{3.31}$$

where last bound is obtained from (3.27) by noting that $\zeta^2 = O(\sigma_0^2 + 48B^2\|\sigma\|_2^2)$ and replacing $\sigma_{X,f}^2 = \sigma_f^2 + D_X^2\|\sigma\|_2^2$. Note that similar bound as in (3.29) has been obtained before by using more complicated algorithms (e.g., penaly method) or different criterrions. On the other hand the complexity bounds in (3.30) and (3.31) appear to be new in the literature. Second, if $B < \|y^*\|_2+1$, then $\mathcal{H}_* > 0$ and as a result, the ConEx method still converges but at the rate of nonsmooth problems in all these three settings described above.

It should be noted that, different from the strongly convex case (c.f. (3.15)), the stepsize scheme in (3.24) depends on $\mathcal{H}_*$, implying that we need to estimate whether $B > \|y*\|_2 + 1$. However, we

can replace $\mathcal{H}_*$ in the definition of η by $\mathcal{H}_B := H_0 + BH_f$. In this way, similar complexity bounds will be obtained for most cases, including nonsmooth deterministic, nonsmooth semi-stochastic, nonsmooth fully-stochastic, as well as smooth semi-stochastic and smooth fully-stochastic problems. In particular, with this modification the last term in (3.27) will change to

$$\tfrac{18}{\varepsilon^2}\big[D_X\sqrt{\mathcal{H}_B^2 + \sigma_0^2 + 48B^2\|\sigma\|_2^2} + \tfrac{D_X(\zeta^2+\mathcal{H}_*^2)}{\sqrt{\mathcal{H}_B^2+\sigma_0^2+48B^2\|\sigma\|_2^2}}\big]^2.$$

The only exception that this modification would not work is for smooth deterministic problems. In this case, since $\mathcal{H}_B = 0$ but $\mathcal{H}_* > 0$, the stepsize scheme (3.24) set according to replacing $\mathcal{H}_*$ by $\mathcal{H}_B$ does not yield convergence. In particular, the last term in the infeasibility bound (3.26) would change to $\mathcal{H}_*^2/\mathcal{H}_B$ which is undefined. One possible solution for this is to artificially set $\mathcal{H}_B > 0$ in the definition of η to be some large positive number and forego of the faster convergence of $O(1/\varepsilon)$. After this change, we would obtain a convergence rate of $O(1/\varepsilon^2)$. An alternative approach would be to design a line search procedure on $\mathcal{H}_B$ for the right value of $\mathcal{H}_*$, since there exists a verifiable condition based on the constraint violation $\|\big[\psi(\cdot)\big]_+\|_2$.

3.4 Convergence analysis of the ConEx method

In this section, we provide a combined analysis of Theorem 3.3.1 and Theorem 3.3.3. Note that Algorithm 1 is essentially a dual type method. In order to analyze this algorithm, we define a *primal-dual gap function* for the equivalent saddle point problem (3.6). In particular, given a pair of feasible solution $z = (x, y)$ and $\bar{z} = (\bar{x}, \bar{y})$ of (3.6), we define the primal-dual gap function $Q(z, \bar{z})$ as

$$Q(z, \bar{z}) := \mathcal{L}(x, \bar{y}) - \mathcal{L}(\bar{x}, y). \tag{3.32}$$

One can easily see from (3.7) that $Q(z, z^*) \geqslant 0$ and $Q(z^*, z) \leqslant 0$ for all feasible z. We use the gap function of the saddle point formulation (3.6) to bound the optimality and feasibility of the convex problem (3.1) separately, in terms of Definition 3.3.1. We first develop an important upper-bound on the gap function in terms of primal, dual variables and randomness. This bound holds for all

nonnegative γ_t, η_t and τ_t. The precise statement is provided in Lemma 3.4.2.

The following technical result provides a simple form of the three-point theorem (see, e.g., Lemma 3.5 of [54]) and will be used in the proof of Lemma 3.4.2.

Lemma 3.4.1 *Assume that $g : X \to \mathbb{R}$ satisfies*

$$g(y) \geqslant g(x) + \langle g'(x), y - x\rangle + \mu W(y, x), \qquad \forall x, y \in S \tag{3.33}$$

for some $\mu \geqslant 0$, where S is convex set in $\mathbb{R}^n$. If

$$\bar{x} = \operatorname*{argmin}_{x \in S} \{g(x) + W(x, \widetilde{x})\},$$

then

$$g(\bar{x}) + W(\bar{x}, \widetilde{x}) + (\mu + 1)W(x, \bar{x}) \leqslant g(x) + W(x, \widetilde{x}), \qquad \forall x \in S.$$

Proof. It follows from the definition of W that $W(x, \widetilde{x}) = W(\bar{x}, \widetilde{x}) + \langle \nabla W(\bar{x}, \widetilde{x}), x - \bar{x}\rangle + W(x, \bar{x})$. Using this relation, (3.33) and the optimality condition for $\bar{x}$, we have

$$\begin{aligned} g(x) + W(x, \widetilde{x}) &= g(x) + [W(\bar{x}, \widetilde{x}) + \langle \nabla W(\bar{x}, \widetilde{x}), x - \bar{x}\rangle + W(x, \bar{x})] \\ &\geqslant g(\bar{x}) + \langle g'(\bar{x}), x - \bar{x}\rangle + \mu W(x, \bar{x}) + [W(\bar{x}, \widetilde{x}) + \langle \nabla W(\bar{x}, \widetilde{x}), x - \bar{x}\rangle + W(x, \bar{x})] \\ &\geqslant g(\bar{x}) + W(\bar{x}, \widetilde{x}) + (\mu + 1)W(x, \bar{x}). \end{aligned}$$

Hence we conclude the proof. □

Lemma 3.4.2 *Suppose* (3.9)*,* (3.10)*,* (3.11) *and* (3.14) *are satisfied. Let $B \geqslant 0$ be a constant and*

assume that $\{\gamma_t, \eta_t, \tau_t, \theta_t\}$ *is a non-negative sequence satisfying*

$$\begin{aligned}
&\gamma_t\theta_t = \gamma_{t-1},\\
&\gamma_t\tau_t \leqslant \gamma_{t-1}\tau_{t-1},\\
&\gamma_t\eta_t \leqslant \gamma_{t-1}(\eta_{t-1} + \alpha_{0,t-1}),
\end{aligned} \tag{3.34}$$

and

$$\begin{aligned}
(2M_f)^2\tfrac{\theta_t}{\theta_{t-1}} &\leqslant \tfrac{\tau_t(\eta_{t-2}-L_0-BL_f)}{12}, \quad \theta_t(M_f+M_\chi)^2 \leqslant \tfrac{\tau_t(\eta_{t-1}-L_0-BL_f)}{12},\\
(2M_f)^2\tfrac{1}{\theta_{T-1}} &\leqslant \tfrac{\tau_t(\eta_{t-2}-L_0-BL_f)}{12}, \qquad (M_f+M_\chi)^2 \leqslant \tfrac{\tau_{T-1}(\eta_{T-1}-L_0-BL_f)}{12},
\end{aligned} \tag{3.35}$$

where $\alpha_{0,t} := \alpha_0 + \alpha^T y_{t+1}$ *and* M_f, M_χ, L_f *are constants as defined in* (3.13)*. Then, for all* $T \geqslant 1$ *and* $z \in \{(x,y) : x \in X, y \geqslant \mathbf{0}\}$*, we have*

$$\begin{aligned}
&\textstyle\sum_{t=0}^{T-1}\gamma_t Q(z_{t+1}, z) + \sum_{t=0}^{T-1}\gamma_t\big[\langle\delta_t^G, x_t - x\rangle - \langle\delta_{t+1}^F, y_{t+1} - y\rangle\big]\\
&\leqslant \gamma_0\eta_0 W(x, x_0) - \gamma_{T-1}(\eta_T + \alpha_{0,T-1})W(x, x_T) + \tfrac{\gamma_0\tau_0}{2}\|y - y_0\|_2^2 - \tfrac{\gamma_{T-1}\tau_{T-1}}{12}\|y - y_T\|_2^2\\
&\quad + \textstyle\sum_{t=0}^{T-1}\frac{2\gamma_t}{\eta_t - L_0 - BL_f}\big[\|\delta_t^G\|_*^2 + (H_0 + H_f\|y\|_2 + \frac{L_f D_X}{2}\big[\|y\|_2 - B\big]_+)^2\big]\\
&\quad + \textstyle\sum_{t=1}^{T-1}\frac{3\gamma_t\theta_t^2}{2\tau_t}\|q_t - \bar{q}_t\|_2^2 + \frac{3\gamma_{T-1}}{2\tau_{T-1}}\|q_T - \bar{q}_T\|_2^2.
\end{aligned} \tag{3.36}$$

Here $q_t := \ell_F(x_t) - \ell_F(x_{t-1}) + \chi(x_t) - \chi(x_{t-1})$*,* $\bar{q}_t := \ell_f(x_t) - \ell_f(x_{t-1}) + \chi(x_t) - \chi(x_{t-1})$*,* $\delta_t^F := \ell_F(x_t) - \ell_f(x_t)$ *and* $\delta_t^G := G_0(x_t, \xi_t) + \sum_{i\in[m]} G_i(x_t)y_{t+1}^{(i)} - f_0'(x_t) - \sum_{i=1}^m f_i'(x_t)y_{t+1}^{(i)}$*.*

Proof. Note that $y_{t+1} = \underset{y\geqslant\mathbf{0}}{\operatorname{argmin}}\langle -s_t, y\rangle + \frac{\tau_t}{2}\|y - y_t\|_2^2$. Hence, using Lemma 3.4.1, we have for all $y \geqslant \mathbf{0}$,

$$-\langle s_t, y_{t+1} - y\rangle \leqslant \tfrac{\tau_t}{2}\big[\|y - y_t\|_2^2 - \|y_{t+1} - y_t\|_2^2 - \|y - y_{t+1}\|_2^2\big]. \tag{3.37}$$

Let us denote $v_t := f_0'(x_t) + \sum_{i\in[m]} f_i'(x_t)y_{t+1}^{(i)}$ and $V_t := G_0(x_t, \xi_t) + \sum_{i\in[m]} G_i(x_t, \xi_t)y_{t+1}^{(i)}$. Then, due to the strong convexity of χ_0 and $\chi_i, i = 1, \ldots, m$, the optimality of x_{t+1}, Lemma 3.4.1 and

the definition of $\alpha_{0,t}$, we have for all $x \in X$,

$$\begin{aligned}\langle V_t, x_{t+1} - x\rangle + \chi_0(x_{t+1}) - \chi_0(x) + \textstyle\sum_{i\in[m]}(\chi_i(x_{t+1}) - \chi_i(x))y_{t+1}^{(i)} \\ \leqslant \eta_t[W(x, x_t) - W(x_{t+1}, x_t)] - (\eta_t + \alpha_{0,t})W(x, x_{t+1}).\end{aligned} \tag{3.38}$$

Due to the convexity of f_0 and f_i, (3.9), the definition of ℓ_f and the fact that $y_{t+1} \geqslant \mathbf{0}$, we have

$$\begin{aligned}\langle v_t, x_{t+1} - x\rangle &= \langle f_0'(x_t) + \textstyle\sum_{i\in[m]} f_i'(x_t)y_{t+1}^{(i)}, x_{t+1} - x\rangle \\ &= \langle f_0'(x_t), x_{t+1} - x_t + x_t - x\rangle + \langle f'(x_t)y_{t+1}, x_{t+1} - x_t + x_t - x\rangle \\ &\geqslant f_0(x_t) - f_0(x) + f_0(x_{t+1}) - f_0(x_t) - \tfrac{L_0}{2}\|x_{t+1} - x_t\|^2 - H_0\|x_{t+1} - x_t\| \\ &\quad + \langle y_{t+1}, \ell_f(x_{t+1}) - f(x_t)\rangle + \langle y_{t+1}, f(x_t) - f(x)\rangle \\ &= f_0(x_{t+1}) - f_0(x) + \langle \ell_f(x_{t+1}) - f(x), y_{t+1}\rangle - \underbrace{\big(\tfrac{L_0}{2}\|x_{t+1} - x_t\|^2 + H_0\|x_{t+1} - x_t\|\big)}_{O_{t+1}},\end{aligned} \tag{3.39}$$

where $O_{t+1} := \frac{L_0}{2}\|x_{t+1} - x_t\|^2 + H_0\|x_{t+1} - x_t\|$ is a 'Lipschitz'-like term for the objective. Combining (3.38), (3.39), noting that $\delta_t^G = V_t - v_t$ and using $\psi_0 = f_0 + \chi_0$, $\psi = f + \chi$, we have

$$\begin{aligned}&\psi_0(x_{t+1}) - \psi_0(x) + \langle \ell_f(x_{t+1}) + \chi(x_{t+1}) - \psi(x), y_{t+1}\rangle + \langle \delta_t^G, x_{t+1} - x\rangle \\ &\leqslant \eta_t W(x, x_t) - \eta_t W(x_{t+1}, x_t) - (\eta_t + \alpha_{0,t})W(x, x_{t+1}) + O_{t+1}.\end{aligned} \tag{3.40}$$

Noting the definition of $Q(\cdot, \cdot)$ in (3.32) and, adding (3.37) and (3.40), we obtain

$$\begin{aligned}Q(z_{t+1}, z) - \langle \psi(x_{t+1}), y\rangle + \langle \ell_f(x_{t+1}) + \chi(x_{t+1}), y_{t+1}\rangle - \langle s_t, y_{t+1} - y\rangle + \langle \delta_t^G, x_{t+1} - x\rangle \\ \leqslant \tfrac{\tau_t}{2}\big[\|y - y_t\|_2^2 - \|y_{t+1} - y_t\|_2^2 - \|y - y_{t+1}\|_2^2\big] \\ + \eta_t W(x, x_t) - \eta_t W(x_{t+1}, x_t) - (\eta_t + \alpha_{0,t})W(x, x_{t+1}) + O_{t+1}.\end{aligned} \tag{3.41}$$

In view of (3.10),

$$f_i(x_{t+1}) - \ell_{f_i}(x_{t+1}) \leqslant \tfrac{L_i}{2}\|x_{t+1} - x_t\|^2 + H_i\|x_{t+1} - x_t\|.$$

Then, using Cauchy-Schwarz inequality and noting definitions of L_f, H_f, we have

$$\langle y, f(x_{t+1}) - \ell_f(x_{t+1})\rangle \leqslant \|y\|_2 \underbrace{\left[\tfrac{L_f}{2}\|x_{t+1} - x_t\|^2 + H_f\|x_{t+1} - x_t\|\right]}_{C_{t+1}},$$

where $C_{t+1} := \frac{L_f}{2}\|x_{t+1} - x_t\|^2 + H_f\|x_{t+1} - x_t\|$ is a 'Lipschitz'-like term for the constraints. Noting the above relation and definitions of q_t and δ^F_{t+1}, we have

$$\begin{aligned}
&\langle \ell_f(x_{t+1}) + \chi(x_{t+1}), y_{t+1}\rangle - \langle \psi(x_{t+1}), y\rangle - \langle s_t, y_{t+1} - y\rangle \\
&\quad \geqslant \langle \ell_f(x_{t+1}) + \chi(x_{t+1}), y_{t+1}\rangle - \langle \ell_f(x_{t+1}) + \chi(x_{t+1}), y\rangle - \langle s_t, y_{t+1} - y\rangle - \|y\|_2 C_{t+1} \\
&\quad = \langle \ell_f(x_{t+1}) + \chi(x_{t+1}) - s_t, y_{t+1} - y\rangle - \|y\|_2 C_{t+1} \\
&\quad = \langle \ell_f(x_{t+1}) + \chi(x_{t+1}) - \ell_F(x_t) - \chi(x_t) - \theta_t q_t, y_{t+1} - y\rangle - \|y\|_2 C_{t+1} \\
&\quad = \langle q_{t+1}, y_{t+1} - y\rangle - \theta_t\langle q_t, y_t - y\rangle - \theta_t\langle q_t, y_{t+1} - y_t\rangle - \langle \delta^F_{t+1}, y_{t+1} - y\rangle - \|y\|_2 C_{t+1}. \quad (3.42)
\end{aligned}$$

Let $B \geqslant 0$ be a constant. Then

$$\begin{aligned}
\|y\|_2 C_{t+1} &= \tfrac{L_f}{2}(\|y\|_2 - B)\|x_{t+1} - x_t\|^2 + \tfrac{BL_f}{2}\|x_{t+1} - x_t\|^2 + \|y\|_2 H_f\|x_{t+1} - x_t\| \\
&\leqslant \tfrac{L_f}{2}[\|y\|_2 - B]_+\|x_{t+1} - x_t\|^2 + \tfrac{BL_f}{2}\|x_{t+1} - x_t\|^2 + \|y\|_2 H_f\|x_{t+1} - x_t\| \\
&\leqslant \tfrac{BL_f}{2}\|x_{t+1} - x_t\|^2 + \big(\|y\|_2 H_f + \tfrac{L_f D_X}{2}[\|y\|_2 - B]_+\big)\|x_{t+1} - x_t\|. \quad (3.43)
\end{aligned}$$

By (3.41), (3.42), and (3.43), noting the definition of O_{t+1} and using the relation $\frac{1}{2}\|a - b\|^2 \leqslant W(a, b)$, we have

$$\begin{aligned}
&Q(z_{t+1}, z) + \langle q_{t+1}, y_{t+1} - y\rangle - \theta_t\langle q_t, y_t - y\rangle + \langle \delta^G_t, x_t - x\rangle - \langle \delta^F_{t+1}, y_{t+1} - y\rangle \\
&\leqslant \theta_t\langle q_t, y_{t+1} - y_t\rangle - \langle \delta^G_t, x_{t+1} - x_t\rangle \\
&\quad + \eta_t W(x, x_t) - (\eta_t + \alpha_{0,t})W(x, x_{t+1}) + \tfrac{\tau_t}{2}\big[\|y - y_t\|_2^2 - \|y_{t+1} - y_t\|_2^2 - \|y - y_{t+1}\|_2^2\big] \\
&\quad - (\eta_t - L_0 - BL_f)W(x_{t+1}, x_t) + \big(H_0 + \|y\|_2 H_f + \tfrac{L_f D_X}{2}[\|y\|_2 - B]_+\big)\|x_{t+1} - x_t\|. \quad (3.44)
\end{aligned}$$

Multiplying (3.44) by γ_t, summing them up from $t = 0$ to $T-1$ with $T \geqslant 1$, we obtain

$$
\begin{aligned}
&\textstyle\sum_{t=0}^{T-1}\gamma_t Q(z_{t+1}, z) + \sum_{t=0}^{T-1}[\gamma_t\langle q_{t+1}, y_{t+1} - y\rangle - \gamma_t\theta_t\langle q_t, y_t - y\rangle] + \sum_{t=0}^{T-1}\gamma_t[\langle \delta_t^G, x_t - x\rangle - \langle \delta_{t+1}^F, y_{t+1} - y\rangle] \\
&\leqslant \textstyle\sum_{t=0}^{T-1}[\gamma_t\theta_t\langle q_t - \bar{q}_t, y_{t+1} - y_t\rangle + \gamma_t\theta_t\langle \bar{q}_t, y_{t+1} - y_t\rangle + \langle \gamma_t\delta_t^G, x_t - x_{t+1}\rangle] \\
&\quad + \textstyle\sum_{t=0}^{T-1}\big[\frac{\gamma_t\tau_t}{2}\|y - y_t\|_2^2 - \frac{\gamma_t\tau_t}{2}\|y - y_{t+1}\|_2^2\big] - \sum_{t=0}^{T-1}\frac{\gamma_t\tau_t}{2}\|y_{t+1} - y_t\|_2^2 \\
&\quad + \textstyle\sum_{t=0}^{T-1}[\gamma_t\eta_t W(x, x_t) - \gamma_t(\eta_t + \alpha_{0,t})W(x, x_{t+1})] \\
&\quad - \textstyle\sum_{t=0}^{T-1}\big[\gamma_t(\eta_t - L_0 - BL_f)W(x_{t+1}, x_t) - \gamma_t \underbrace{\big(H_0 + \|y\|_2 H_f + \frac{L_f D_X}{2}[\|y\|_2 - B]_+\big)}_{\mathcal{H}(y,B)}\|x_{t+1} - x_t\|\big],
\end{aligned}
\tag{3.45}
$$

where $\mathcal{H}(y, B) := H_0 + \|y\|_2 H_f + \frac{L_f D_X}{2}[\|y\|_2 - B]_+$. Now we focus our attention to handle the inner product terms of (3.45). Noting the definition of $\bar{q}_t$, we have

$$
\begin{aligned}
\|\bar{q}_t\|_2 &= \|\ell_f(x_t) - \ell_f(x_{t-1}) + \chi(x_t) - \chi(x_{t-1})\|_2 \\
&\leqslant \|f(x_{t-1}) + f'(x_{t-1})^T(x_t - x_{t-1}) - f(x_{t-2}) - f'(x_{t-2})^T(x_{t-1} - x_{t-2})\|_2 + \|\chi(x_t) - \chi(x_{t-1})\|_2 \\
&\leqslant \|f(x_{t-1}) - f(x_{t-2})\|_2 + \|f'(x_{t-1})^T(x_t - x_{t-1})\|_2 + \|f'(x_{t-2})^T(x_{t-1} - x_{t-2})\|_2 + M_H\|x_t - x_{t-1}\| \\
&\leqslant 2M_f\|x_{t-1} - x_{t-2}\| + (M_f + M_H)\|x_t - x_{t-1}\|,
\end{aligned}
\tag{3.46}
$$

where the last relation follows due to (3.12). Using the above relation, we obtain

$$
\begin{aligned}
&\gamma_t\theta_t\langle \bar{q}_t, y_{t+1} - y_t\rangle - \tfrac{\gamma_t\tau_t}{3}\|y_{t+1} - y_t\|_2^2 - \tfrac{\gamma_{t-2}(\eta_{t-2} - L_0 - BL_f)}{4}W(x_{t-1}, x_{t-2}) - \tfrac{\gamma_{t-1}(\eta_{t-1} - L_0 - BL_f)}{4}W(x_t, x_{t-1}) \\
&\quad \leqslant \gamma_t\theta_t\|\bar{q}_t\|_2\|y_{t+1} - y_t\|_2 - \tfrac{\gamma_t\tau_t}{3}\|y_{t+1} - y_t\|_2^2 \\
&\qquad - \tfrac{\gamma_{t-2}(\eta_{t-2} - L_0 - BL_f)}{4}W(x_{t-1}, x_{t-2}) - \tfrac{\gamma_{t-1}(\eta_{t-1} - L_0 - BL_f)}{4}W(x_t, x_{t-1}) \\
&\quad \leqslant 2M_f\gamma_t\theta_t\|x_{t-1} - x_{t-2}\|\|y_{t+1} - y_t\|_2 - \tfrac{\gamma_t\tau_t}{6}\|y_{t+1} - y_t\|_2^2 - \tfrac{\gamma_{t-2}(\eta_{t-2} - L_0 - BL_f)}{4}W(x_{t-1}, x_{t-2}) \\
&\qquad + (M_f + M_H)\gamma_t\theta_t\|x_t - x_{t-1}\|\|y_{t+1} - y_t\|_2 - \tfrac{\gamma_t\tau_t}{6}\|y_{t+1} - y_t\|_2^2 - \tfrac{\gamma_{t-1}(\eta_{t-1} - L_0 - BL_f)}{4}W(x_t, x_{t-1}) \\
&\quad \leqslant 0,
\end{aligned}
\tag{3.47}
$$

where the last inequality follows by applying the relation $W(x,y) \geqslant \frac{1}{2}\|x-y\|$, Young's inequality ($2ab \leqslant a^2+b^2$) applied twice, once with

$$a = \left(\tfrac{\gamma_t\tau_t}{6}\right)^{1/2}\|y_{t+1}-y_t\|, \qquad b = \left(\tfrac{\gamma_{t-2}(\eta_{t-2}-L_0-BL_f)}{8}\right)^{1/2}\|x_{t-1}-x_{t-2}\|$$

and second time with

$$a = \left(\tfrac{\gamma_t\tau_t}{6}\right)^{1/2}\|y_{t+1}-y_t\|, \qquad b = \left(\tfrac{\gamma_{t-1}(\eta_{t-1}-L_0-BL_f)}{8}\right)^{1/2}\|x_t-x_{t-1}\|,$$

and the fact that

$$\begin{aligned}
(2M_f)\gamma_t\theta_t \leqslant \left\{\tfrac{\gamma_t\gamma_{t-2}\tau_t(\eta_{t-2}-L_0-BL_f)}{12}\right\}^{1/2} \quad &\Leftrightarrow \qquad (2M_f)^2\tfrac{\theta_t}{\theta_{t-1}} \leqslant \tfrac{\tau_t(\eta_{t-2}-L_0-BL_f)}{12},\\
(M_f+M_H)^2\gamma_t^2\theta_t^2 \leqslant \tfrac{\gamma_t\gamma_{t-1}\tau_t(\eta_{t-1}-L_0-BL_f)}{12} \quad &\Leftrightarrow \quad (M_f+M_H)^2\theta_t \leqslant \tfrac{\tau_t(\eta_{t-1}-L_0-BL_f)}{12},
\end{aligned}$$

where equivalences follow due to (3.34).

Using Young's inequality, Cauchy-Schwarz inequality and the relation $u^Tv \leqslant \|u\|\|v\|_*$, we have

$$\begin{aligned}
\gamma_t\theta_t\langle q_t-\bar{q}_t, y_{t+1}-y_t\rangle - \tfrac{\gamma_t\tau_t}{6}\|y_{t+1}-y_t\|_2^2 &\leqslant \tfrac{3\gamma_t\theta_t^2}{2\tau_t}\|q_t-\bar{q}_t\|_2^2,\\
\langle\gamma_t\delta_t^G, x_t-x_{t+1}\rangle - \tfrac{\gamma_t(\eta_t-L_0-BL_f)}{4}W(x_{t+1},x_t) &\leqslant \tfrac{2\gamma_t}{\eta_t-L_0-BL_f}\|\delta_t^G\|_*^2,\\
\gamma_t\mathcal{H}(y,B)\|x_{t+1}-x_t\| - \tfrac{\gamma_t(\eta_t-L_0-BL_f)}{4}W(x_{t+1},x_t) &\leqslant \tfrac{2\gamma_t}{\eta_t-L_0-BL_f}\mathcal{H}(y,B)^2.
\end{aligned} \tag{3.48}$$

Using (3.47) and (3.48) for $t=0,\ldots,T-1$ inside (3.45) and noting (3.34), we have

$$\begin{aligned}
&\textstyle\sum_{t=0}^{T-1}\gamma_tQ(z_{t+1},z) + \gamma_{T-1}\langle q_T, y_T-y\rangle + \sum_{t=0}^{T-1}\gamma_t[\langle\delta_t^G, x_t-x\rangle - \langle\delta_{t+1}^F, y_{t+1}-y\rangle]\\
&\quad\leqslant \gamma_0\eta_0W(x,x_0) - \gamma_{T-1}(\eta_t+\alpha_{0,T-1})W(x,x_T) + \tfrac{\gamma_0\tau_0}{2}\|y-y_0\|_2^2 - \tfrac{\gamma_{T-1}\tau_{T-1}}{2}\|y-y_T\|_2^2\\
&\qquad+\textstyle\sum_{t=0}^{T-1}\Big[\tfrac{3\gamma_t\theta_t^2}{2\tau_t}\|q_t-\bar{q}_t\|_2^2 + \tfrac{2\gamma_t}{\eta_t-L_0-BL_f}\|\delta_t^G\|_*^2 + \tfrac{2\gamma_t}{\eta_t-L_0-BL_f}\mathcal{H}(y,B)^2\Big]\\
&\qquad-\tfrac{\gamma_{T-2}(\eta_{T-2}-L_0-BL_f)}{4}W(x_{T-1},x_{T-2}) - \tfrac{\gamma_{T-1}(\eta_{T-1}-L_0-BL_f)}{2}W(x_T,x_{T-1}),
\end{aligned} \tag{3.49}$$

where in the left hand side of the above relation, we used the fact that $q_0 = \ell_F(x_0) - \ell_F(x_{-1}) + \chi(x_0) - \chi(x_{-1}) = \mathbf{0}$. Similarly, we see that $\bar{q}_0 = \mathbf{0}$. Hence we can ignore $\|q_0-\bar{q}_0\|_2^2$ term in the right hand side of the above relation.

Using (3.46), we have

$$
\begin{aligned}
&- \gamma_{T-1}\langle \bar{q}_T, y_T - y\rangle - \tfrac{\gamma_{T-1}\tau_{T-1}}{3}\|y - y_T\|_2^2 \\
&\quad - \tfrac{\gamma_{T-2}(\eta_{T-2}-L_0-BL_f)}{4} W(x_{T-1}, x_{T-2}) - \tfrac{\gamma_{T-1}(\eta_{T-1}-L_0-BL_f)}{2} W(x_T, x_{T-1}) \\
&\leqslant (M_f + M_H)\gamma_{T-1}\|x_T - x_{T-1}\|\,\|y_T - y\|_2 - \tfrac{\gamma_{T-1}\tau_{T-1}}{12}\|y - y_T\|_2^2 - \tfrac{\gamma_{T-1}(\eta_{T-1}-L_0-BL_f)}{2} W(x_T, x_{T-1}) \\
&\quad + 2M_f\gamma_{T-1}\|x_{T-1} - x_{T-2}\|\,\|y_T - y\|_2 - \tfrac{\gamma_{T-1}\tau_{T-1}}{6}\|y - y_T\|_2^2 - \tfrac{\gamma_{T-2}(\eta_{T-2}-L_0-BL_f)}{4} W(x_{T-1}, x_{T-2}) \\
&\quad - \tfrac{\gamma_{T-1}\tau_{T-1}}{12}\|y_T - y\|_2^2 \\
&\leqslant -\tfrac{\gamma_{T-1}\tau_{T-1}}{12}\|y_T - y\|_2^2, \qquad (3.50)
\end{aligned}
$$

where the last relation follows from (3.35), Young's inequality and the fact that

$$
\begin{aligned}
(2M_f)\gamma_{T-1} \leqslant \left\{\tfrac{\gamma_{T-2}\gamma_{T-1}\tau_{T-1}(\eta_{T-2}-L_0-BL_f)}{12}\right\}^{1/2} \quad &\Leftrightarrow \quad (2M_f)^2\tfrac{1}{\theta_{T-1}} \leqslant \tfrac{\tau_t(\eta_{t-2}-L_0-BL_f)}{12}, \\
(M_f + M_H)\gamma_{T-1} \leqslant \left\{\tfrac{\gamma_{T-1}^2\tau_{T-1}(\eta_{T-1}-L_0-BL_f)}{12}\right\}^{1/2} \quad &\Leftrightarrow \quad (M_f + M_H)^2 \leqslant \tfrac{\tau_{T-1}(\eta_{T-1}-L_0-BL_f)}{12}.
\end{aligned}
$$

Moreover, again using Young's inequality and Cauchy-Schwarz inequality, we have

$$
-\gamma_{T-1}\langle q_T - \bar{q}_T, y_T - y\rangle - \tfrac{\gamma_{T-1}\tau_{T-1}}{6}\|y - y_T\|_2^2 \leqslant \tfrac{3\gamma_{T-1}}{2\tau_{T-1}}\|q_T - \bar{q}_T\|_2^2. \qquad (3.51)
$$

Using (3.50) and (3.51) in relation (3.49), noting that $q_0 - \bar{q}_0 = \mathbf{0}$ and replacing the definition of $\mathcal{H}(y, B)$, we obtain (3.36). □

We now aim to convert the bound on the primal-dual gap function Q in Lemma 3.4.2 into a bound on the optimality and infeasibility according to Definition 3.3.1. For proving this lemma, we need one more simple result which is stated below.

Lemma 3.4.3 *Let $\rho_0, \ldots, \rho_j$ be a sequence of elements in $\mathbb{R}^n$ and let S be a convex set in $\mathbb{R}^n$. Define the sequence $v_t, t = 0, 1, \ldots,$ as follows: $v_0 \in S$ and*

$$
v_{t+1} = \operatorname*{argmin}_{x \in S} \langle \rho_t, x\rangle + \tfrac{1}{2}\|x - v_t\|_2^2.
$$

Then for any $x \in S$ and $t \geqslant 0$, the following inequalities hold

$$\langle \rho_t, v_t - x \rangle \leqslant \tfrac{1}{2}\|x - v_t\|_2^2 - \tfrac{1}{2}\|x - v_{t+1}\|_2^2 + \tfrac{1}{2}\|\rho_t\|_2^2, \tag{3.52}$$

$$\textstyle\sum_{t=0}^{j} \langle \rho_t, v_t - x \rangle \leqslant \tfrac{1}{2}\|x - v_0\|_2^2 + \tfrac{1}{2}\sum_{t=0}^{j}\|\rho_t\|_2^2. \tag{3.53}$$

Proof. Using Lemma 3.4.1 with $g(x) = \langle \rho_t, x \rangle$, $W(y, x) = \frac{1}{2}\|y - x\|_2^2$, $\widetilde{x} = v_t$ and $\mu = 0$, we have, due to the optimality of v_{t+1},

$$\langle \rho_t, v_{t+1} - x \rangle + \tfrac{1}{2}\|v_{t+1} - v_t\|_2^2 + \tfrac{1}{2}\|x - v_{t+1}\|_2^2 \leqslant \tfrac{1}{2}\|x - v_t\|_2^2,$$

is satisfied for all $x \in S$. The above relation and the fact

$$\langle \rho_t, v_t - v_{t+1} \rangle - \tfrac{1}{2}\|v_{t+1} - v_t\|_2^2 \leqslant \tfrac{1}{2}\|\rho_t\|_2^2,$$

imply that

$$\langle \rho_t, v_t - x \rangle \leqslant \tfrac{1}{2}\|x - v_t\|_2^2 - \tfrac{1}{2}\|x - v_{t+1}\|_2^2 + \tfrac{1}{2}\|\rho_t\|_2^2,$$

for all $x \in S$. Summing up the above relations from $t = 0$ to j and noting the nonnegativity of $\|\cdot\|_2^2$, we obtain (3.53). Hence we conclude the proof. □

Now we are ready to prove the lemma converting bound on the primal-dual gap to infeasibility and optimality gap.

Lemma 3.4.4 *Suppose all assumptions in Lemma 3.4.2 are satisfied. Then, for $T \geqslant 1$, we have*

$$\begin{aligned}\mathbb{E}[\psi_0(\overline{x}_T) - \psi_0(x^*)] \leqslant{}& \tfrac{1}{\Gamma_T}\big[\gamma_0\eta_0 W(x^*, x_0) + \tfrac{\gamma_0\tau_0}{2}\|y_0\|_2^2 \\ &+ \textstyle\sum_{t=0}^{T-1} \frac{2\gamma_t}{\eta_t - L_0 - BL_f}\big(\mathbb{E}[\|\delta_t^G\|_*^2] + H_0^2\big) + \big(\sum_{t=1}^{T-1} \frac{12\gamma_t\theta_t^2}{\tau_t} + \frac{12\gamma_{T-1}}{\tau_{T-1}}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2)\big],\end{aligned} \tag{3.54}$$

$$\gamma_{T-1}(\eta_{T-1}+\alpha_{0,T-1})\mathbb{E}[W(x^*,x_T)] \leqslant \tfrac{\gamma_0\tau_0}{2}\|y^*-y_0\|_2^2 + \gamma_0\eta_0 W(x^*,x_0)$$
$$+ \big(\textstyle\sum_{t=1}^{T-1}\frac{12\gamma_t\theta_t^2}{\tau_t} + \frac{12\gamma_{T-1}}{\tau_{T-1}}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2) \tag{3.55}$$
$$+ \textstyle\sum_{t=0}^{T-1}\frac{2\gamma_t}{\eta_t - L_0 - BL_f}\big\{\mathbb{E}[\|\delta_t^G\|_*^2] + (H_0 + \|y^*\|_2 H_f + [\|y^*\|_2 - B]_+)^2\big\},$$

and

$$\mathbb{E}[\|\big[\psi(\bar{x}_T)\big]_+\|_2] \leqslant \tfrac{1}{\Gamma_T}\Big[\gamma_0\tau_0\|y_0\|_2^2 + 3(\|y^*\|_2+1)^2\gamma_0\tau_0 + \gamma_0\eta_0 W(x^*,x_0)$$
$$+ \textstyle\sum_{t=0}^{T-1}\frac{2\gamma_t}{\eta_t - L_0 - BL_f}\big[\mathbb{E}[\|\delta_t^G\|_*^2] + (H_0 + (\|y^*\|_2+1)H_f + \frac{L_f D_X[\|y^*\|_2+1-B]_+}{2})^2\big]$$
$$+ \big(\textstyle\sum_{t=1}^{T-1}\frac{12\gamma_t\theta_t^2}{\tau_t} + \sum_{t=0}^{T-1}\frac{\gamma_t}{\tau_t} + \frac{12\gamma_{T-1}}{\tau_{T-1}}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2)\Big]. \tag{3.56}$$

where $\Gamma_T := \sum_{t=0}^{T-1}\gamma_t$

Proof. Notice that conditional random variables $[G_0(x_t,\xi_t)|\xi_{[t-1]},\bar{\xi}_{[t-2]}]$ and $[G_i(x_t,\xi_t)|\xi_{[t-1]},\bar{\xi}_{[t-2]}]$ satisfy properties of SO in (3.14) because x_t is a constant conditioned on random variables $\xi_{[t-1]} := (\xi_0,\ldots,\xi_{t-1})$ and $\bar{\xi}_{[t-2]} := (\bar{\xi}_0,\ldots,\bar{\xi}_{t-2})$. Also, observe that, y_{t+1} is a constant conditioned on random variables $\xi_{[t-1]}$ and $\bar{\xi}_{[t-1]}$. In particular, using (3.14), we have

$$\mathbb{E}[\langle\delta_t^G, x_t - x\rangle] = \mathbb{E}\langle\mathbb{E}_{|\xi_{[t-1]},\bar{\xi}_{[t-1]}}[\delta_t^G], x_t - x\rangle = 0, \tag{3.57}$$

for any non-random x. This follows due to the following relation

$$\mathbb{E}_{|\xi_{[t-1]},\bar{\xi}_{[t-1]}}[\delta_t^G]$$
$$= \mathbb{E}_{|\xi_{[t-1]},\bar{\xi}_{[t-1]}}[G_0(x_t,\xi_t) - f_0'(x_t)] + \mathbb{E}\textstyle\sum_{i=1}^m y_{t+1}^{(i)}\mathbb{E}_{|\xi_{[t-1]},\bar{\xi}_{[t-1]}}[G_i(x_t,\xi_t) - f_i'(x_t)] = \mathbf{0}.$$

Similarly, using (3.14), we have

$$\mathbb{E}[\big\langle\delta_{t+1}^F, y_{t+1} - y\big\rangle] = \mathbb{E}[\big\langle\mathbb{E}_{|\xi_{[t]},\bar{\xi}_{[t-1]}}[\delta_{t+1}^F], y_{t+1} - y\big\rangle] = 0, \tag{3.58}$$

for any non-random y. Here, we note that

$$\begin{aligned}\mathbb{E}_{|\xi_{[t]},\bar{\xi}_{[t-1]}}[\delta_{t+1}^F] = \mathbb{E}_{|\xi_{[t]},\bar{\xi}_{[t-1]}}[F(x_t,\bar{\xi}_t)] - f(x_t) \\ + \big(\mathbb{E}_{|\xi_{[t]},\bar{\xi}_{[t-1]}}[\mathbf{G}(x_t,\bar{\xi}_t)] - f'(x_t)\big)^T(x_{t+1}-x_t) = \mathbf{0},\end{aligned} \tag{3.59}$$

where the first term in RHS is $\mathbf{0}$ due to the third relation in (3.14) applied to $\bar{\xi}_t$, the second term is $\mathbf{0}$ due to the second relation of (3.14) applied to $\bar{\xi}_t$ and the common fact for both the terms that x_t, x_{t+1} are constants for given $\xi_{[t]}, \bar{\xi}_{[t-1]}$. We note that

$$\begin{aligned}\mathbb{E}[\|\delta_t^F\|_2^2] &\leqslant 2\mathbb{E}[\|F(x_{t-1},\bar{\xi}_{t-1}) - f(x_{t-1})\|_2^2] + 2\mathbb{E}[\|[\mathbf{G}(x_{t-1},\bar{\xi}_{t-1}) - f'(x_{t-1})]^T(x_t - x_{t-1})\|_2^2] \\ &\leqslant 2\sigma_f^2 + 2\mathbb{E}[\textstyle\sum_{i=1}^m \big\{(G_i(x_{t-1},\bar{\xi}_{t-1}) - f_i'(x_{t-1}))^T(x_t - x_{t-1})\big\}^2] \\ &\leqslant 2\sigma_f^2 + 2\mathbb{E}[\textstyle\sum_{i=1}^m \|G_i(x_{t-1},\bar{\xi}_{t-1}) - f'(x_{t-1})\|_*^2 \|x_t - x_{t-1}\|^2] \\ &\leqslant 2\sigma_f^2 + 2D_X^2\|\sigma\|_2^2.\end{aligned} \tag{3.60}$$

Then, in view of above relation and definitions of $q_t, \bar{q}_t$, we have

$$\begin{aligned}\mathbb{E}[\|q_t - \bar{q}_t\|_2^2] &= \mathbb{E}[\|\ell_F(x_t) - \ell_f(x_t) + \ell_F(x_{t-1}) - \ell_f(x_{t-1})\|_2^2] \\ &\leqslant 2\mathbb{E}[\|\delta_t^F\|_2^2] + 2\mathbb{E}[\|\delta_{t-1}^F\|_2^2] \leqslant 8(\sigma_f^2 + D_X^2\|\sigma\|_2^2).\end{aligned} \tag{3.61}$$

Taking expectation on both sides of (3.36) and using relation (3.57), (3.58) and (3.61), we have for all non-random[2] $z \in \{(x,y) : x \in X, y \geqslant \mathbf{0}\}$,

$$\begin{aligned}\mathbb{E}[\textstyle\sum_{t=0}^{T-1}\gamma_t Q(z_{t+1},z)] \leqslant \tfrac{\gamma_0\tau_0}{2}\|y-y_0\|_2^2 + \gamma_0\eta_0 W(x,x_0) + \big(\textstyle\sum_{t=1}^{T-1}\frac{12\gamma_t\theta_t^2}{\tau_t} + \frac{12\gamma_{T-1}}{\tau_{T-1}}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2) \\ + \textstyle\sum_{t=0}^{T-1}\frac{2\gamma_t}{\eta_t - L_0 - BL_f}\big[\mathbb{E}[\|\delta_t^G\|_*^2] + (H_0 + \|y\|_2 H_f + \frac{L_f D_X[\|y\|_2 - B]_+}{2})^2\big] \\ - \gamma_{T-1}(\eta_{T-1} + \alpha_{0,T-1})\mathbb{E}[W(x,x_T)],\end{aligned} \tag{3.62}$$

where we dropped $\|y - y_T\|_2^2$. Using the convexity of $\psi_0(\cdot)$ and $\psi(\cdot)$, and noting the definition of

[2]This x, y is required to be non-random because we are dropping the inner product terms of the left hand side of (3.36).

Γ_T, we have for all non-random $y \geqslant \mathbf{0}$ and $x \in X$,

$$\Gamma_T \mathbb{E}\big[\psi_0(\bar{x}_T) + \langle y, \psi(\bar{x}_T)\rangle - \psi_0(x) - \langle \bar{y}_T, \psi(x)\rangle\big] \leqslant \mathbb{E}[\textstyle\sum_{t=0}^{T-1} \gamma_t Q(z_{t+1}, z)]. \tag{3.63}$$

Combining (3.62) and (3.63), then choosing $x = x^*, y = \mathbf{0}$ (which are non-random) throughout the combined relation, observing that $[0 - B]_+ = 0$ for any $B \geqslant 0$, ignoring $W(x, x_T)$ term and noting that $\psi(x^*) \leqslant \mathbf{0}$ and $\bar{y}_T \geqslant \mathbf{0}$ implies $\langle \bar{y}_T, \psi(x^*)\rangle \leqslant 0$, we have (3.54).

Now, we prove a bound on $\mathbb{E}[W(x^*, x_T)]$. Put $z = z^* := (x^*, y^*)$ in (3.62). Then we have that $Q(z_{t+1}, z^*) \geqslant 0$ for all $t = 0, \ldots, T-1$. Hence, using $z = z^*$ in (3.62), dropping summation of Q-terms and taking expectation on both sides, we obtain (3.55).

Now, we focus our attention to the infeasibility bound. First, define $R := \|y^*\|_2 + 1$. Second, define an auxiliary sequence $\{y_t^v\}$ in the following way: $y_0^v = y_0$ and for all $t \geqslant 0$, define

$$y_{t+1}^v := \underset{y \in \mathcal{B}_+^2(R)}{\operatorname{argmin}} \tfrac{1}{\tau_{t-1}} \langle \delta_t^F, y\rangle + \tfrac{1}{2}\|y - y_t^v\|_2^2,$$

where we recall that $\mathcal{B}_+^2(R) = \{x \in \mathbb{R}^n : \|x\|_2 \leqslant R, x \geqslant \mathbf{0}\}$. Then in view of Lemma 3.4.3, in particular relation (3.52), for all $y \in \mathcal{B}_+^2(R)$ we have

$$\tfrac{1}{\tau_t}\langle \delta_{t+1}^F, y_{t+1}^v - y\rangle \leqslant \tfrac{1}{2}\|y - y_{t+1}^v\|_2^2 - \tfrac{1}{2}\|y - y_{t+2}^v\|_2^2 + \tfrac{1}{2\tau_t^2}\|\delta_{t+1}^F\|_2^2. \tag{3.64}$$

Multiplying (3.64) by $\gamma_t\tau_t$, taking a sum from $t = 0$ to $T-1$ and noting the second relation in (3.34), we obtain

$$\textstyle\sum_{t=0}^{T-1} \gamma_t \langle \delta_{t+1}^F, y_{t+1}^v - y\rangle \leqslant \frac{\gamma_0\tau_0}{2}\|y - y_1^v\|_2^2 + \sum_{t=0}^{T-1} \frac{\gamma_t}{2\tau_t}\|\delta_{t+1}^F\|_2^2, \tag{3.65}$$

for all $y \in \mathcal{B}_+^2(R)$. Summing (3.65) and (3.36), we obtain

$$\begin{aligned}
&\textstyle\sum_{t=0}^{T-1}\gamma_t Q(z_{t+1},z) + \sum_{t=0}^{T-1}\gamma_t[\langle \delta_t^G, x_t - x\rangle - \langle \delta_{t+1}^F, y_{t+1} - y_{t+1}^v\rangle] \\
&\leqslant \tfrac{\gamma_0\tau_0}{2}[\|y-y_0\|_2^2 + \|y-y_1^v\|_2^2] + \gamma_0\eta_0 W(x,x_0) + \textstyle\sum_{t=1}^{T-1}\frac{3\gamma_t\theta_t^2}{2\tau_t}\|q_t-\bar{q}_t\|_2^2 + \frac{3\gamma_{T-1}}{2\tau_{T-1}}\|q_T-\bar{q}_T\|_2^2 \\
&\quad + \textstyle\sum_{t=0}^{T-1}\big[\frac{2\gamma_t}{\eta_t - L_0 - BL_f}\{\|\delta_t^G\|_*^2 + (H_0 + \|y\|_2 H_f + \frac{L_f D_X[\|y\|_2 - B]_+}{2})^2\} + \frac{\gamma_t}{2\tau_t}\|\delta_{t+1}^F\|_2^2\big], \qquad (3.66)
\end{aligned}$$

for all $z \in \big\{(x,y) : x \in X, y \in \mathcal{B}_+^2(R)\big\}$. Note that given $\xi_{[t]}$ and $\bar{\xi}_{[t-1]}$, we have $y_{t+1}, y_{t+1}^v, x_{t+1}$ and x_t are constants. Hence we have

$$\mathbb{E}[\langle \delta_{t+1}^F, y_{t+1} - y_{t+1}^v\rangle] = \mathbb{E}[\langle \mathbb{E}_{|\xi_{[t]},\bar{\xi}_{[t-1]}}[\delta_{t+1}^F], y_{t+1} - y_{t+1}^v\rangle] = 0, \qquad (3.67)$$

where second equality follows from (3.59). Choosing $z = \widehat{z} := (x^*, \widehat{y})$ in (3.66) where $\widehat{y} := (\|y^*\|_2 + 1)\big[\psi(\bar{x}_T)\big]_+ \big\|\big[\psi(\bar{x}_T)\big]_+\big\|_2^{-1} \in \mathcal{B}_+^2(R)$, taking expectation on both sides and noting (3.67), (3.60), (3.61), first relation in (3.57), we have

$$\begin{aligned}
\mathbb{E}[\textstyle\sum_{t=0}^{T-1}\gamma_t Q(z_{t+1},\widehat{z})] &\leqslant \tfrac{\gamma_0\tau_0}{2}\mathbb{E}[\|\widehat{y}-y_0\|_2^2 + \|\widehat{y}-y_1^v\|_2^2] + \gamma_0\eta_0 W(x^*,x_0) \\
&\quad + \textstyle\sum_{t=0}^{T-1}\frac{2\gamma_t}{\eta_t - L_0 - BL_f}\big\{\mathbb{E}[\|\delta_t^G\|_*^2] + \big(H_0 + (\|y^*\|_2 + 1)H_f + \frac{L_f D_X[\|y^*\|_2 + 1 - B]_+}{2}\big)^2\big\} \\
&\quad + \big(\textstyle\sum_{t=1}^{T-1}\frac{12\gamma_t\theta_t^2}{\tau_t} + \sum_{t=0}^{T-1}\frac{\gamma_t}{\tau_t} + \frac{12\gamma_{T-1}}{\tau_{T-1}}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2). \qquad (3.68)
\end{aligned}$$

Noting the convexity of Q in first argument, we obtain

$$\mathbb{E}[Q(\bar{z}_T,\widehat{z})] \leqslant \tfrac{1}{\Gamma_T}\mathbb{E}[\textstyle\sum_{t=0}^{T-1}\gamma_t Q(z_{t+1},\widehat{z})]. \qquad (3.69)$$

Now observe that

$$\begin{aligned}
&\mathcal{L}(\bar{x}_T, y^*) - \mathcal{L}(x^*, y^*) \geqslant 0 \\
\Rightarrow\ &\psi_0(\bar{x}_T) + \langle y^*, \psi(\bar{x}_T)\rangle - \psi_0(x^*) \geqslant 0,
\end{aligned}$$

which in view of the relation

$$\langle y^*, \psi(\bar{x}_T)\rangle \leqslant \langle y^*, \big[\psi(\bar{x}_T)\big]_+\rangle \leqslant \|y^*\|_2 \|\big[\psi(\bar{x}_T)\big]_+\|_2,$$

implies that

$$\psi_0(\bar{x}_T) + \|y^*\|_2 \|\big[\psi(\bar{x}_T)\big]_+\|_2 - \psi_0(x^*) \geqslant 0. \tag{3.70}$$

Moreover,

$$Q(\bar{z}_T, \widehat{z}) = \mathcal{L}(\bar{x}_T, \widehat{y}) - \mathcal{L}(x^*, \bar{y}_T) \geqslant \mathcal{L}(\bar{x}_T, \widehat{y}) - \mathcal{L}(x^*, y^*) = \psi_0(\bar{x}_T) + (\|y^*\|_2 + 1)\|\big[\psi(\bar{x}_T)\big]_+\|_2 - \psi_0(x^*),$$

along with (3.70) implies that

$$Q(\bar{z}_T, \widehat{z}) \geqslant \|\big[\psi(\bar{x}_T)\big]_+\|_2.$$

The above relation, (3.69) and (3.68) together yield

$$\begin{aligned}\mathbb{E}[\|\big[\psi(\bar{x}_T)\big]_+\|_2] \leqslant \tfrac{1}{\Gamma_T}\Big[& \tfrac{\gamma_0\tau_0}{2}\mathbb{E}[\|\widehat{y} - y_0\|_2^2 + \|\widehat{y} - y_1^v\|_2^2] + \gamma_0\eta_0 W(x^*, x_0) \\ & + \textstyle\sum_{t=0}^{T-1} \frac{2\gamma_t}{\eta_t - L_0 - BL_f}\big\{\mathbb{E}[\|\delta_t^G\|_*^2] + \big(H_0 + (\|y^*\|_2 + 1)H_f + \frac{L_f D_X[\|y^*\|_2 + 1 - B]_+}{2}\big)^2\big\} \\ & + \big(\textstyle\sum_{t=1}^{T-1} \frac{12\gamma_t\theta_t^2}{\tau_t} + \sum_{t=0}^{T-1}\frac{\gamma_t}{\tau_t} + \frac{12\gamma_{T-1}}{\tau_{T-1}}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2)\Big].\end{aligned}$$

Noting the bound $\|\widehat{y} - y_1^v\|_2 \leqslant 2R$ and $\|\widehat{y} - y_0\|_2^2 \leqslant 2\|y_0\|_2^2 + 2\|\widehat{y}\|_2^2 \leqslant \|y_0\|_2^2 + 2R^2$ in the above relation and recalling that $R = \|y^*\|_2 + 1$, we obtain (3.56). Hence we conclude the proof. □

Note that we still need to bound $\mathbb{E}[\|\delta_t^G\|_*^2]$. Below, we provide a simple lemma which is used to show such a bound.

Lemma 3.4.5 *Let $\{a_t\}_{t\geqslant 0}$ be a nonnegative sequence, $m_1, m_2 \geqslant 0$ be constants such that $a_0 \leqslant m_1$ and the following relation holds for all $t \geqslant 1$:*

$$a_t \leqslant m_1 + m_2 \textstyle\sum_{k=0}^{t-1} a_k.$$

Then we have $a_t \leqslant m_1(1+m_2)^t$.

Proof. We prove this lemma by induction. Clearly, it is true for $t = 0$. Suppose it is true for a_t. Then, using inductive hypothesis on a_k for $k = 0, \ldots, t$, we have

$$\begin{aligned} a_{t+1} &\leqslant m_1 + m_2 \textstyle\sum_{k=0}^{t} a_t \\ &\leqslant m_1\big[1 + m_2 \textstyle\sum_{k=0}^{t}(1+m_2)^k\big] \\ &\leqslant m_1\big[1 + m_2 \tfrac{(1+m_2)^{t+1}-1}{m_2}\big] = m_1(1+m_2)^{t+1}. \end{aligned}$$

Hence, we conclude the proof. □

Now, under some assumptions, we show a bound on $\mathbb{E}[\|\delta_t^G\|_*^2]$.

Lemma 3.4.6 *Assume that* $\{\gamma_t, \tau_t, \eta_t\}$ *satisfy*

$$\tfrac{96\|\sigma\|_2^2}{\tau_t(\eta_t - L_0 - BL_f)} < 1 \tag{3.71}$$

for all $t \leqslant T-1$ *and constants* R_1 *and* R_2 *satisfying the following conditions exist.*

$$\begin{aligned} R_1 \geqslant \Big(1 - \tfrac{96\|\sigma\|_2^2}{\tau_t(\eta_t - L_0 - BL_f)}\Big)^{-1}\Big[2\sigma_0^2 + \tfrac{48\|\sigma\|_2^2}{\gamma_t\tau_t}\Big\{\gamma_0\eta_0 W(x^*, x_0) + \tfrac{\gamma_0\tau_0}{2}\|y^* - y_0\|_2^2 + \tfrac{\gamma_t\tau_t}{12}\|y*\|_2^2 \\ + \textstyle\sum_{i=0}^{t} \tfrac{2\gamma_i}{\eta_i - L_0 - BL_f}\big(H_0 + H_f\|y^*\|_2 + \tfrac{L_f D_X[\|y^*\|_2 - B]_+}{2}\big)^2 \\ + \big(\textstyle\sum_{i=1}^{t} \tfrac{12\gamma_i\theta_i^2}{\tau_i} + \tfrac{12\gamma_t}{\tau_t}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2)\Big\}\Big] \end{aligned} \tag{3.72}$$

for all $t \leqslant T-1$ *and*

$$R_2 \geqslant \Big(1 - \tfrac{96\|\sigma\|_2^2}{\tau_t(\eta_t - L_0 - BL_f)}\Big)^{-1} \tfrac{96\|\sigma\|_2^2\gamma_i}{\gamma_t\tau_t(\eta_i - L_0 - BL_f)} \tag{3.73}$$

for all $t \leqslant T-1$ *and* $i \leqslant t-1$. *Then, we have*

$$\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant R_1(1+R_2)^t, \tag{3.74}$$

for all $t \leqslant T-1$. *In particular, if* $\|\sigma\|_2 = 0$*, then we can set* $R_1 = 2\sigma_0^2$ *and* $R_2 = 0$ *implying* $\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant 2\sigma_0^2$.

Proof. Observe that $Q(z_{t+1}, z^*) \geqslant 0$ for all $t = 0, \ldots, T-1$ where $z^* = (x^*, y^*)$. Choosing $z = z^*$ in (3.36) for T substituted by $t+1 (\geqslant 1)$, taking expectation, using (3.57) with $x = x^*$ and (3.58) with $y = y^*$ and noting (3.61), we have

$$\begin{aligned}
\tfrac{\gamma_t\tau_t}{12}\mathbb{E}\|y^* - y_{t+1}\|_2^2 &\leqslant \gamma_0\eta_0 W(x^*, x_0) + \tfrac{\gamma_0\tau_0}{2}\|y^* - y_0\|_2^2 \\
&+ \textstyle\sum_{i=0}^t \frac{2\gamma_i}{\eta_i - L_0 - BL_f}\big[\mathbb{E}\|\delta_i^G\|_*^2 + (H_0 + H_f\|y^*\|_2 + \frac{L_f D_X[\|y^*\|_2 - B]_+}{2})^2\big] \\
&+ \big(\textstyle\sum_{i=1}^t \frac{12\gamma_i\theta_i^2}{\tau_i} + \frac{12\gamma_t}{\tau_t}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2). \qquad (3.75)
\end{aligned}$$

Now, let us define $\delta_{t,i}^G := G_i(x_t, \xi_t) - f_i'(x_t)$ for $i = 0, \ldots, m$. As a consequence, we have $\delta_t^G = \delta_{t,0}^G + \sum_{i=1}^m y_{t+1}^{(i)}\delta_{t,i}^G$. Then, we have

$$\begin{aligned}
\mathbb{E}[\|\delta_t^G\|_*^2] &= \mathbb{E}[\|\delta_{t,0}^G + \textstyle\sum_{i=1}^m y_{t+1}^{(i)}\delta_{t,i}^G\|_*^2] \\
&\overset{\text{(i)}}{\leqslant} 2\mathbb{E}[\|\delta_{t,0}^G\|_*^2] + 2\mathbb{E}\big[\|\textstyle\sum_{i=1}^m y_{t+1}^{(i)}\delta_{t,i}^G\|_*^2\big] \\
&\leqslant 2\mathbb{E}[\|\delta_{t,0}^G\|_*^2] + 2\mathbb{E}\big[(\textstyle\sum_{i=1}^m \|y_{t+1}^{(i)}\delta_{t,i}^G\|_*)^2\big] \\
&\overset{\text{(ii)}}{\leqslant} 2\big\{\sigma_0^2 + \mathbb{E}\big[\|y_{t+1}\|_2^2\big(\textstyle\sum_{i=1}^m \|\delta_{t,i}^G\|_*^2\big)\big]\big\} \\
&\overset{\text{(iii)}}{\leqslant} 2\big\{\sigma_0^2 + \mathbb{E}\big[\|y_{t+1}\|_2^2\big(\textstyle\sum_{i=1}^m \mathbb{E}_{|\xi_{[t-1]}, \bar{\xi}_{[t-1]}}[\|\delta_{t,i}^G\|_*^2]\big)\big]\big\} \\
&\overset{\text{(iv)}}{\leqslant} 2\big\{\sigma_0^2 + \mathbb{E}[\|y_{t+1}\|_2^2 \textstyle\sum_{i=1}^m \sigma_i^2]\big\} \\
&= 2(\sigma_0^2 + \|\sigma\|_2^2\mathbb{E}\|y_{t+1}\|_2^2) \\
&\leqslant 2\sigma_0^2 + 4\|\sigma\|_2^2\big(\|y^*\|_2^2 + \mathbb{E}\|y_{t+1} - y^*\|_2^2\big). \qquad (3.76)
\end{aligned}$$

Here, relation (i) follows due to the fact that $\|a+b\|_*^2 \leqslant (\|a\|_* + \|b\|_*)^2 \leqslant 2\|a\|_*^2 + 2\|b\|_*^2$, relation (ii) follows due to Cauchy-Schwarz inequality, relation (iii) follows dues the fact that y_{t+1} is a constant conditioned on random variables $\xi_{[t-1]}, \bar{\xi}_{[t-1]}$ and relation (iv) follows from fourth and fifth relation in (3.14) and the fact that x_t is a constant conditioned on random variables $\xi_{[t-1]}, \bar{\xi}_{[t-1]}$.

Adding $\frac{\gamma_t\tau_t}{12}\|y^*\|_2^2$ to both sides of (3.75), then multiplying it by $\frac{48\|\sigma\|_2^2}{\gamma_t\tau_t}$ and observing (3.76), we have

$$\begin{aligned}\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant 2\sigma_0^2 + \tfrac{48\|\sigma\|_2^2}{\gamma_t\tau_t}\Big\{\gamma_0\eta_0 W(x^*,x_0) + \tfrac{\gamma_0\tau_0}{2}\|y^*-y_0\|_2^2 + \tfrac{\gamma_t\tau_t}{12}\|y^*\|_2^2 \\ + \textstyle\sum_{i=0}^t \frac{2\gamma_i}{\eta_i - L_0 - BL_f}(H_0 + H_f\|y^*\|_2 + \frac{L_f D_X[\|y^*\|_2 - B]_+}{2})^2 \\ + \big(\textstyle\sum_{i=1}^t \frac{12\gamma_i\theta_i^2}{\tau_i} + \frac{12\gamma_t}{\tau_t}\big)(\sigma_f^2 + D_X^2\|\sigma\|_2^2)\Big\} + \textstyle\sum_{i=0}^t \frac{96\|\sigma\|_2^2\gamma_i}{\gamma_t\tau_t(\eta_i - L_0 - BL_f)}\mathbb{E}\|\delta_i^G\|_*^2.\end{aligned}$$

In view of (3.71), we have that the coefficient of the δ_t^G term on the right hand side of the above relation is strictly less than 1. Moving the δ_t^G term to the left hand side and noting the conditions imposed on constants R_1, R_2, we have

$$\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant R_1 + R_2\textstyle\sum_{i=0}^{t-1}\mathbb{E}[\|\delta_i^G\|_*^2],$$

for all $t \leqslant T-1$. Using Lemma 3.4.5 for the above relation, we have (3.74). Hence we conclude the proof. □

Note that bound in (3.74) is still a function of stepsize parameters since R_1 are R_2 need to satisfy relations (3.72) and (3.73), respectively. Now, we need to show that there exists a possible selection of stepsize parameters for which we can compute a uniform upper bound on $\mathbb{E}[\|\delta_t^G\|_*^2]$ for all $t \leqslant T-1$, in particular, we can obtain constants R_1 and R_2 satisfying (3.72) and (3.73), respectively. Moreover, selected stepsize policy is meaningful in the sense that it yields convergence according (3.54) and (3.56). Below, we show that the stepsize policy in (3.15) of Theorem 3.3.1 and (3.24) of Theorem 3.3.3 are specified in a way such that (3.34), (3.35) and (3.71) are satisfied. Moreover, a uniform upper bound according to (3.74) for all $t \leqslant T-1$ can be obtained and it also leads to the convergence according to (3.54) and (3.56). In particular, we show the proof of Theorem 3.3.1 and Theorem 3.3.3 below.

First, we focus on the setting in which (3.1) is strongly convex, i.e., $\alpha_0 > 0$ and show the proof of Theorem 3.3.1 below.

Proof of Theorem 3.3.1. Note that $\{\gamma_t, \theta_t, \eta_t, \tau_t\}$ set according to (3.15) satisfy (3.34). It is easy to verify the first two relations in (3.34). To verify the third relation, note that

$$\begin{aligned}\gamma_{t-1}(\eta_{t-1} + \alpha_{0,t-1}) &\geqslant \gamma_{t-1}(\eta_{t-1} + \alpha_0) \\ &= (t + t_0 + 1)\big(\tfrac{\alpha_0(t+t_0)}{2} + \alpha_0\big) = \tfrac{\alpha_0}{2}(t + t_0 + 1)(t + t_0 + 2) = \gamma_t\eta_t.\end{aligned}$$

Note that (3.35) is satisfied if $\frac{4}{3}\mathcal{M}^2 \leqslant \frac{\tau_t(\eta_{t-2}-L_0-BL_f)}{12}$. This follows due to the fact that $\{\eta_t\}$ is an increasing sequence, $\frac{3}{4} \leqslant \theta_t < 1$ and the definition of $\mathcal{M}$. Indeed we have,

$$\tfrac{\tau_t(\eta_{t-2}-L_0-BL_f)}{12} \geqslant \tfrac{32\mathcal{M}^2}{12\alpha_0(t+1)}\big(\tfrac{\alpha_0(t+t_0-1)}{2} - \tfrac{\alpha_0(t_0-2)}{4}\big) = \tfrac{2(2t+t_0)\mathcal{M}^2}{3(t+1)} \geqslant \tfrac{4\mathcal{M}^2}{3},$$

where the last inequality follows from $t_0 \geqslant 2$ by definition. Also note that

$$\tau_t(\eta_t - L_0 - BL_f) \geqslant \tfrac{384\|\sigma\|_2^2T}{\alpha_0(t+1)}\big(\tfrac{\alpha_0(t+t_0+1)}{2} - \tfrac{\alpha_0(t_0-2)}{4}\big) = \tfrac{96(2t+t_0+4)\|\sigma\|_2^2T}{t+1} \geqslant 192\|\sigma\|_2^2$$

for all $t \geqslant 0$. Then the above relation implies that

$$\tfrac{96\|\sigma\|_2^2}{\tau_t(\eta_t-L_0-BL_f)} \leqslant \tfrac{1}{2}, \tag{3.77}$$

for all $t \geqslant 0$. Finally, we need to show the existence of constants R_1 and R_2 satisfying (3.72) and (3.73), respectively. Using the fact that $\tau_t \geqslant \frac{384\|\sigma\|_2^2T}{\alpha_0(t+1)}$, we observe

$$\tfrac{96\|\sigma\|_2^2\gamma_i}{\gamma_t\tau_t(\eta_i-L_0-BL_f)} \leqslant \tfrac{384\|\sigma\|_2^2(i+t_0+2)}{\alpha_0(2i+t_0+4)}\tfrac{\alpha_0(t+1)}{384\|\sigma\|_2^2(t+t_0+2)T} \leqslant \tfrac{1}{T}, \tag{3.78}$$

for all $i \geqslant 0, t \geqslant 0$. Noting (3.77), (3.78) and (3.73), we can set

$$R_2 := \tfrac{2}{T}. \tag{3.79}$$

Noting (3.72) along with definition of $\mathcal{H}_*$ in the theorem statement, setting $y_0 = \mathbf{0}$, using (3.77),(3.61),

and applying the following relations

$$\begin{aligned}
\gamma_t\tau_t &\geqslant \max\big\{\tfrac{384\|\sigma\|_2^2T}{\alpha_0}, \tfrac{\sigma_{X,f}T^{3/2}}{B(t_0+2)^{1/2}}\big\},\\
\textstyle\sum_{i=0}^t \frac{\gamma_i}{\eta_i - L_0 - BL_f} &\leqslant \tfrac{4(t+1)}{\alpha_0},\\
\textstyle\sum_{i=1}^t \frac{\gamma_i\theta_i^2}{\tau_i} + \frac{\gamma_t}{\tau_t} &\leqslant \tfrac{B(t_0+2)^{1/2}}{\sigma_{X,f}T^{3/2}}\big[\tfrac{(t+1)^3}{3} + \tfrac{(t+1)^2(t_0+2)}{2} + \tfrac{(t+1)(9t_0+10)}{6} - (t_0+1)\big],
\end{aligned}$$

we can observe that have for all $t \leqslant T-1$, RHS of (3.72) is at most

$$\begin{aligned}
&2\Big[2\sigma_0^2 + 48\|\sigma\|_2^2\Big\{\big(\tfrac{t_0+2}{2} + \tfrac{1}{12}\big)\|y^*\|_2^2 + \tfrac{8T\mathcal{H}_*^2}{\alpha_0}\tfrac{T}{T+t_0+1}\tfrac{\alpha_0}{384\|\sigma\|_2^2T}\\
&\quad + \tfrac{12\sigma_{X,f}^2B(t_0+2)^{1/2}}{\sigma_{X,f}T^{3/2}}\Big(\tfrac{B(t_0+2)^{1/2}}{\sigma_{X,f}T^{3/2}}\tfrac{T^3}{3} + \tfrac{\alpha_0}{384\|\sigma\|_2^2T}\big(\tfrac{T^2(t_0+2)}{2} + \tfrac{T(9t_0+10)}{6} - (t_0+1)\big)\Big)\Big\}\Big].
\end{aligned}$$

Then, noting $\frac{1}{T} \leqslant 1$ and ignoring $-(t_0+1)$ term, we can set

$$R_1 := 2\Big[2\sigma_0^2 + 24(t_0+3)\|\sigma\|_2^2\|y^*\|_2^2 + \mathcal{H}_*^2 + 4\times 48(t_0+2)B^2\|\sigma\|_2^2 + 3\alpha_0B\sigma_{X,f}(t_0+2)^{3/2}\Big]. \quad (3.80)$$

Then using Lemma 3.4.6 and noting (3.79), we have for all $t \leqslant T-1$

$$\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant \begin{cases} 2\sigma_0^2 & \text{if } \|\sigma\|_2 = \sigma_f = 0;\\ R_1\big(1+\frac{2}{T}\big)^{T-1} \leqslant R_1e^2 & \text{otherwise.}\end{cases}.$$

Noting the above relation, (3.80) and the definition of ζ, we have

$$\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant \zeta^2, \qquad \forall\, t \leqslant T-1. \quad (3.81)$$

So according to (3.54) with $y_0 = \mathbf{0}$ and using (3.81), we have

$$\begin{aligned}
\mathbb{E}[\psi_0(\overline{x}_T) - \psi_0(x^*)] \leqslant \tfrac{2}{T(T+2t_0+3)}\big[&\tfrac{\alpha_h(t_0+1)(t_0+2)}{2}W(x^*,x_0) + \tfrac{8(\zeta^2+H_0^2)T}{\alpha_0}\\
&+ 12B(t_0+2)^{1/2}\sigma_{X,f}\big\{\tfrac{T^{1/2}(T+2)}{3} + \tfrac{(t_0+1)T^{-1/2}(T+3)}{2}\big\}\big].
\end{aligned}$$

Here we used the bound

$$\begin{aligned} \tfrac{\gamma_t}{\eta_t - L_0 - BL_f} &\leqslant \tfrac{4}{\alpha_0} \text{ for all } t \geqslant 0, \\ \textstyle\sum_{t=1}^{T-1} \tfrac{\gamma_t \theta_t^2}{\tau_t} + \tfrac{\gamma_{T-1}}{\tau_{T-1}} &\leqslant \tfrac{B(t_0+2)^{1/2}}{\sigma_{X,f} T^{3/2}} \big[\tfrac{T^2(T+2)}{3} + (t_0+1)\tfrac{T(T+3)}{2} \big]. \end{aligned} \tag{3.82}$$

Noting the bound on $W(x^*, x_0)$ in the earlier relation, we obtain (3.16). Using (3.56), (3.81) and the bounds in (3.82), we have

$$\begin{aligned} \mathbb{E} \Big\| \big[\psi(\bar{x}_T) \big]_+ \Big\|_2 \leqslant \tfrac{2}{T(T+2t_0+3)} \big[& 3(t_0+2)(\|y^*\|_2 + 1)^2 \max\big\{ \tfrac{32\mathcal{M}^2}{\alpha_0}, \tfrac{\sigma_{X,f} T^{3/2}}{B(t_0+2)^{1/2}}, \tfrac{384 \|\sigma\|_2^2 T}{\alpha_0} \big\} \\ & + \tfrac{\alpha_0 (t_0+1)(t_0+2)}{2} W(x^*, x_0) + 13B(t_0+2)^{1/2} \sigma_{X,f} \big\{ \tfrac{T^{1/2}(T+2)}{3} + \tfrac{(t_0+1)T^{-1/2}(T+3)}{2} \big\} \\ & + \tfrac{8T}{\alpha_0} \big\{ \zeta^2 + \big[H_0 + (\|y^*\|_2 + 1) H_f + \tfrac{L_f D_X [\|y^*\|_2 + 1 - B]_+}{2} \big]^2 \big\} \big]. \end{aligned} \tag{3.83}$$

Noting the bound on $W(x^*, x_0)$ in (3.83), the definition of $\mathcal{H}_*$, using the fact that $\frac{T^{1/2}(T+2)}{3} \leqslant T^{3/2}$ and combining the $T^{3/2}$ order terms, we obtain (3.17). From (3.55), we have

$$\begin{aligned} \mathbb{E}[W(x_T, x^*)] \leqslant \tfrac{2}{\alpha_0 (T+t_0+1)(T+t_0+2)} \big[& \tfrac{(t_0+2)\|y^*\|_2^2}{2} \max\big\{ \tfrac{32\mathcal{M}^2}{\alpha_0}, \tfrac{\sigma_{X,f} T^{3/2}}{B(t_0+2)^{1/2}}, \tfrac{384 \|\sigma\|_2^2 T}{\alpha_0} \big\} \\ & + \tfrac{\alpha_0 (t_0+1)(t_0+2)}{2} W(x^*, x_0) + 12B(t_0+2)^{1/2} \sigma_{X,f} \big\{ \tfrac{T^{1/2}(T+2)}{3} + \tfrac{(t_0+1)T^{-1/2}(T+3)}{2} \big\} \\ & + \tfrac{8T}{\alpha_0} \big\{ \zeta^2 + \big[H_0 + \|y^*\|_2 H_f + \tfrac{L_f D_X [\|y^*\|_2 - B]_+}{2} \big]^2 \big\} \big]. \end{aligned}$$

With similar replacements in the above relation as in (3.83), we obtain (3.18). Hence we conclude the proof. □

Proof of Theorem 3.3.3. It is easy to verify that $\{\gamma_t, \theta_t, \eta_t, \tau_t\}$ set according to (3.24) satisfy (3.34) with $\alpha_0 = 0$. Note that (3.35) is satisfied if $\mathcal{M}^2 \leqslant \frac{\tau_t(\eta_{t-2} - L_0 - BL_f)}{12}$. This follows due to the fact that $\{\eta_t\}$ is an non-decreasing sequence, $\theta_t = 1$ for all $t \geqslant 0$ and the definition of $\mathcal{M}$. Then we have

$$\tfrac{\tau_t(\eta_{t-2} - L_0 - BL_f)}{12} \geqslant \tfrac{6\mathcal{M}B}{D_X} \tfrac{2\mathcal{M}D_X}{B} \times \tfrac{1}{12} = \mathcal{M}^2.$$

Also, since $(\eta_t - L_0 - BL_f) \geqslant \frac{24B\|\sigma\|_2}{D_X}$ and $\tau_t \geqslant \frac{8D_X\|\sigma\|_2}{B}$, we have

$$\tau_t(\eta_t - L_0 - BL_f) \geqslant 192\|\sigma\|_2^2$$

for all $t \geqslant 0$. In view of the above relation, we have

$$\tfrac{96\|\sigma\|_2^2}{\tau_t(\eta_t - L_0 - BL_f)} \leqslant \tfrac{1}{2}, \tag{3.84}$$

hence (3.71) is satisfied. We also need to show the existence of R_1 and R_2 satisfying (3.72) and (3.73), respectively. Using the fact that γ_t, η_t and τ_t are constants for all $t \geqslant 0$, $\tau\eta \geqslant \frac{96T\sigma_{X,f}\|\sigma\|_2}{D_X}$ and noting (3.84), we obtain

$$\big(1 - \tfrac{96\|\sigma\|_2^2}{\tau_t(\eta_t - L_0 - BL_f)}\big)^{-1} \tfrac{96\|\sigma\|_2^2\gamma_i}{\gamma_t\tau_t(\eta_i - L_0 - BL_f)} \leqslant 2\tfrac{96\|\sigma\|_2^2}{\tau\eta} \leqslant 2\tfrac{\|\sigma\|_2 D_X}{T\sigma_{X,f}} \leqslant \tfrac{2}{T},$$

where in the last relation, we used the fact that $\sigma_{X,f} \geqslant D_X\|\sigma\|_2$. In view of the above relation and (3.73), we can set

$$R_2 := \tfrac{2}{T}. \tag{3.85}$$

Noting (3.72) along with the fact that $\mathcal{H}_* \geqslant H_0 + H_f\|y^*\|_2 + \frac{L_f D_X[\|y^*\|_2 - B]_+}{2}$, setting $y_0 = \mathbf{0}$, using (3.84), (3.61), $\gamma_t\tau_t = \tau \geqslant \sqrt{96T}\sigma_{X,f}$, $\sum_{i=0}^t \frac{\gamma_i}{\eta_i - L_0 - BL_f} = \frac{t+1}{\eta} \leqslant \frac{\sqrt{T}D_X}{\sqrt{2[\mathcal{H}_*^2 + \sigma_0^2 + 48B^2\|\sigma\|_2^2]}}$, and $\sum_{i=1}^t \frac{\gamma_i\theta_i^2}{\tau_i} + \frac{\gamma_t}{\tau_t} = \frac{t+1}{\tau} \leqslant \frac{T}{\tau}$ for all $t \leqslant T-1$, we can see that the RHS of (3.72) is at most

$$\begin{aligned}
&2\big[2\sigma_0^2 + 48\|\sigma\|_2^2\big\{\tfrac{7}{12}\|y^*\|_2^2 + \tfrac{\eta}{\tau}D_X^2 + \tfrac{\sqrt{2T}D_X\mathcal{H}_*^2}{\sqrt{\mathcal{H}_*^2 + \sigma_0^2 + 48B^2\|\sigma\|_2^2}}\tfrac{B}{\sqrt{96T}\sigma_{X,f}} + 12\sigma_{X,f}^2\tfrac{T}{\tau^2}\big\}\big] \\
&\leqslant 2\big[2\sigma_0^2 + 48\|\sigma\|_2^2\big\{\tfrac{7}{12}\|y^*\|_2^2 + \tfrac{\eta}{\tau}D_X^2 + \tfrac{D_X B\mathcal{H}_*}{\sqrt{48}\sigma_{X,f}} + 12T\sigma_{X,f}^2\tfrac{B^2}{96T\sigma_{X,f}^2}\big\}\big] \\
&\leqslant 2\big[2\sigma_0^2 + 48\|\sigma\|_2^2\big\{\tfrac{7}{12}\|y^*\|_2^2 + \tfrac{D_X}{\sigma_{X,f}}\big(B\sqrt{\tfrac{[\mathcal{H}_*^2 + \sigma_0^2 + 48B^2\|\sigma\|_2^2]}{48}} + \tfrac{B\mathcal{H}_*}{\sqrt{48}}\big) + \tfrac{6\max\{\mathcal{M},4\|\sigma\|_2\}BD_X}{2\max\{\mathcal{M},4\|\sigma\|_2\}}\tfrac{B}{D_X} + \tfrac{B^2}{8}\big\}\big] \\
&\leqslant 2\big[2\sigma_0^2 + 28\|\sigma\|_2^2\|y^*\|_2^2 + 75B^2\|\sigma\|_2^2 + \sqrt{48}\|\sigma\|_2[2B\mathcal{H}_* + (B\sigma_0 + \sqrt{48}B^2\|\sigma\|_2)]\big]
\end{aligned}$$

where in the last inequality, we used the fact that $\frac{\|\sigma\|_2 D_X}{\sigma_{X,f}} \leqslant 1$. Note that the last term in the above

sequence of relations is a constant satisfying the requirement in (3.72). Hence we can set

$$R_1 := 2\big[2\sigma_0^2 + 28\|\sigma\|_2^2\|y^*\|_2^2 + 75B^2\|\sigma\|_2^2 + \sqrt{48}\|\sigma\|_2[2B\mathcal{H}_* + (B\sigma_0 + \sqrt{48}B^2\|\sigma\|_2)]\big]. \quad (3.86)$$

Then using Lemma 3.4.6 and noting (3.85), we have for all $t \leqslant T-1$

$$\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant \begin{cases} 2\sigma_0^2 & \text{if } \|\sigma\|_2 = \sigma_f = 0; \\ R_1\big(1+\frac{2}{T}\big)^{T-1} \leqslant R_1 e^2 & \text{otherwise.} \end{cases}.$$

Noting the above relation, (3.86) and the definition of ζ, we have

$$\mathbb{E}[\|\delta_t^G\|_*^2] \leqslant \zeta^2, \qquad \forall\, t \leqslant T-1. \quad (3.87)$$

So according to (3.54) with $y_0 = \mathbf{0}$ and using (3.87), we have

$$\mathbb{E}[\psi_0(\overline{x}_T) - \psi_0(x^*)] \leqslant \tfrac{1}{T}\big[(\eta + L_0 + BL_f)W(x^*, x_0) + \tfrac{2(\zeta^2+H_0^2)T}{\eta} + 12\sigma_{X,f}^2\tfrac{T}{\tau}\big].$$

Using the bound $W(x^*, x_0) \leqslant D_X^2$, we obtain (3.25). From (3.56) and (3.87), we have for $T \geqslant 1$

$$\mathbb{E}\Big\|\big[\psi(\overline{x}_T)\big]_+\Big\|_2 \leqslant \tfrac{1}{T}\big[3(\|y^*\|_2+1)^2\tau + (\eta + L_0 + BL_f)W(x^*, x_0) + \tfrac{2(\zeta^2+\mathcal{H}_*^2)T}{\eta} + \tfrac{13\sigma_{X,f}^2 T}{\tau}\big].$$

Using bounds $W(x^*, x_0) \leqslant D_X^2$, we obtain (3.26). Using (3.26) and (3.27), we have

$$\mathbb{E}\Big\|\big[\psi(\overline{x}_T)\big]_+\Big\|_2 \leqslant \tfrac{\varepsilon}{3} + \tfrac{\varepsilon}{3} + \tfrac{\varepsilon}{3} = \varepsilon,$$

Similarly, using (3.25) and (3.27), it is easy to observe that $\mathbb{E}[\psi_0(\overline{x}_T) - \psi_0(x^*)] \leqslant \varepsilon$. Hence we conclude the proof. □

CHAPTER 4

STOCHASTIC PROXIMAL POINT METHOD FOR STRUCTURED NONCONVEX FUNCTION CONSTRAINED OPTIMIZATION

In the previous chapter, we looked at ConEx method as a unified algorithm for solving the convex composite function constrained optimization problem. In this chapter, we will look at the proximal point method for nonconvex function constrained optimization. We assume that nonconvex functions have a minimal structure such that the original problem can be reduced to solving a sequence of convex composite function constrained subproblems. The algorithm and the analysis techniques are motivated by proximal point methods for unconstrained optimization. We will look at the convergence of the newly proposed proximal point method to the KKT point under various constraint qualifications. We will also consider stochastic or large-scale cases where an exact solution to the convex subproblems cannot be obtained. We will employ the aforementioned ConEx method for solving the subproblems inexactly and show its convergence under various constraint qualifications.

4.1 Structured Nonconvex Function Constrained Optimization

We study the following composite optimization problem with function constraints:

$$\begin{aligned} \min_{x\in X} \quad & \psi_0(x) := f_0(x) + \chi_0(x) \\ \text{s.t.} \quad & \psi_i(x) := f_i(x) + \chi_i(x) \leqslant 0, \quad i = 1, \dots, m, \end{aligned} \tag{4.1}$$

where $X \subseteq \mathbb{R}^n$ is a convex compact set, $f_0 : X \to \mathbb{R}$ and $f_i : X \to \mathbb{R},\ i = 1, \dots, m$ are continuous functions which are not necessarily convex, $\chi_0 : X \to \mathbb{R}$ is a proper convex lower semicontinuous function, and $\chi_i : X \to \mathbb{R}, i = 1, \dots, m$ are convex and continuous functions. Problem 4.1 covers different nonconvex settings depending on the assumptions on f_i and χ_i, $i = 0, \dots, m$.

We assume that f_i, $i = 0, \ldots, m$, are smooth functions, which are not necessarily convex, but satisfying a certain lower curvature condition (c.f. (4.2)). However, we do not put the simplicity assumption about the proximal operator associated with convex functions χ_i, $i = 0, \ldots, m$, as we did in the previous chapter, covering a broader class of nonconvex problems. This includes problems with non-differentiable objective functions or constraints.

4.1.1 Algorithms in the literature

The past few years has seen a resurgence of interest in the design of efficient algorithms for nonconvex stochastic optimization, especially for stochastic and finite-sum problems due to their importance in machine learning. Most of these studies need to assume that the constraints are convex, and focus on the analysis of iteration complexity, i.e., the number of iterations required to find an approximate stationary point, as well as possible ways to accelerate such approximate solutions.

If the nonconvex function constraints do not appear, one type of approach for solving (4.1) is to directly generalize stochastic gradient descent type methods (see [39, 41, 93, 1, 36, 123, 109, 123, 109, 88, 54]) for solving problems with nonconvex objective functions. An alternative approach is to indirectly utilize convex optimization methods within the framework of proximal-point methods which transfer nonconvex optimization problems into a series of convex ones (see [45, 13, 37, 27, 51, 60, 91, 85]). While direct methods are simpler and hence easier to implement, indirect methods may provide stronger theoretical performance guarantees under certain circumstances, e.g., when the problem has a large conditional number, many components and/or multiple blocks [60].

However, if nonconvex function constraints $\psi_i(x) \leqslant 0$ do appear in (4.1), the study on its solution methods is scarce. While there is a large body of work on the asymptotic analysis and the optimality conditions of penalty-based approaches for general constrained nonlinear programming (for example, see [12, 74, 4, 3, 30]), only a few works discussed the complexity of these methods for solving problems with nonconvex function constraints [21, 108, 34]. However, these techniques are not applicable to our setting because they cannot guarantee the feasibility of the generated solutions, but a certain local non-increasing properties for the constraint functions. On the other hand,

the feasibility of the nonconvex function constraints appear to be important in certain problems of interest.

4.1.2 New method for solving structured nonconvex function constrained optimization

In this chapter, we aim to extend the ConEx method for the nonconvex setting and present a new framework of proximal point method for solving the nonconvex function constrained optimization problems, which otherwise seem to be difficult to solve by using direct approaches.

The key component of our method is to exploit the structure of the nonconvex objective and constraints ψ_i, $i = 0, \ldots, m$, thereby turning the original problem into a sequence of function constrained subproblems with a strongly convex objective and strongly convex constraints. We show that when the initial point is strictly feasible, then all the subsequent points generated in the algorithm remain strictly feasible. Hence by Slater condition, there exists Lagrange multipliers attaining strong duality for each subproblem. Furthermore, we analyze the conditions under which the dual variables are bounded, and show asymptotic convergence of the sequence to the KKT points of the original problem. Moreover, we provide the first iteration complexity of this proximal point method under certain regularity conditions. More specifically, we show that this method requires $O(1/\varepsilon)$ iterations to obtain an appropriately defined ε-KKT point.

For practical use, we propose an inexact proximal point type algorithm for which only approximate solutions of the subproblems are given. To develop the convergence analysis of the proposed method, we present different termination criterions for controlling the accuracy for solving the subproblems, either based on the distance to the optimal solution, or in terms of function optimality gap and constraint violation, depending on different types of constraint qualifications. We then establish the convergence or complexity of the inexact proximal point method for solving nonconvex function constrained problems. We also present the overall complexity of the inexact proximal point method when the ConEx method is used to solve the subproblems under appropriate constraint qualification conditions (see Theorem 4.2.14, Corollary 4.2.16 and discussions afterwards).

Almost at the same time this work was completed, Ma et. al. [70] also worked independently on the analysis of the proximal-point methods for nonconvex function constrained problems. In spite of some overlap, there exist a few essential differences between this work and [70]. First, this work establishes the convergence/complexity of the proximal point method under a variety of constraint qualification conditions, including Mangasarian-Fromovitz constraint qualification (MFCQ), strong MFCQ, and strong feasibility, and hence covers a broader class of nonconvex problems, while [70] only consider a uniform Slater's condition. Strong feasibility condition is stronger than the uniform Slater's condition but is easier to verify. Second, [70] uses a different definition of subdifferential than the one proposed here and the definition of the KKT conditions in [70] comes from convex optimization problems. While it is unclear under what constraint qualification this KKT condition is necessary for local optimality of nonsmooth nonconvex problems they consider, it is possible to put their problem into our structured composite framework in 3.1 and compute the subdifferential that provably yields our KKT condition under the aforementioned MFCQ. Third, for solving the convex subproblems, we will use ConEx method presented in Chapter 3, that can achieve the best-known rate of convergence for solving different problem classes, including deterministic, semi-stochastic and fully-stochastic, smooth and nonsmooth problems. On the other hand, different methods were suggested for solving different types of problems in [70]. In particular, a variant of the switching subgradient method, which was firstly presented by Polyak in [90] for the general convex case, and later extended by [62] for the stochastic and strongly convex cases, was suggested for solving deterministic problems. For the stochastic case they directly apply the algorithm in [117] and hence require stochastic gradients to be bounded. These nonsmooth subgradient methods do not necessarily yield the best possible rate of convergence if the objective/constraint functions are smooth or contain certain smooth components.

Now we shift our focus to the details of proximal point method for structured nonconvex function constrained optimization.

4.1.3 Notation and terminologies

We borrow the useful notation in (3.2) from Chapter 3 and the constraints in (3.1) be expressed as $\psi(x) \leqslant \mathbf{0}$. $\|\cdot\|$ denotes a general norm and $\|\cdot\|_*$ denotes its dual norm defined as $\|z\|_* := \sup\{z^T x : \|x\| \leqslant 1\}$. From this definition, we obtain the $a^T b \leqslant \|a\|\|b\|_*$. Euclidean norm is denoted as $\|\cdot\|_2$ and standard inner product is denoted as $\langle\cdot,\cdot\rangle$. Let $\mathcal{B}^2(r) := \{x : \|x\|_2 \leqslant r\}$ be the Euclidean ball of radius r centered at origin. Nonnegative orthant of this ball is denoted as $\mathcal{B}^2_+(r)$. For a convex set X, we denote the normal cone at $x \in X$ as $N_X(x)$ and its dual cone as $N^*_X(x)$, interior as $\operatorname{int} X$ and relative interior as $\operatorname{rint} X$. For a scalar valued function f and a scalar t, the notation $\{f \leqslant t\}$ stands for the set $\{x : f(x) \leqslant t\}$. The "+" operation on sets denotes the Minkowski sum of the sets. We refer to the distance between two sets $A, B \subset \mathbb{R}^n$ as $d(A,B) := \min_{a\in A, b\in B}\|a-b\|$.

$\left[x\right]_+ := \max\{x, 0\}$ for any $x \in \mathbb{R}$. For any vector $x \in \mathbb{R}^k$, we define $\left[x\right]_+$ as elementwise application of the operator $\left[\,\cdot\,\right]_+$. The i-th element of vector x is denoted as x_i unless otherwise explicitly specified a different notation for certain special vectors.

A function $r(\cdot)$ is λ-*Lipschitz smooth* if the gradient $\nabla r(x)$ is a λ-Lipschitz function, i.e. for some $\lambda \geqslant 0$

$$\|\nabla r(x) - \nabla r(y)\|_* \leqslant \lambda\|x-y\|, \;\; \forall x, y \in \operatorname{dom} r.$$

An equivalent form is:

$$-\tfrac{\lambda}{2}\|x-y\|^2 \leqslant r(x) - r(y) - \langle\nabla r(y), x-y\rangle \leqslant \tfrac{\lambda}{2}\|x-y\|^2, \qquad \forall x, y \in \operatorname{dom} r.$$

A refined version of the above property differentiates between negative and positive curvature. In particular, we have

$$r(y) + \langle\nabla r(y), x-y\rangle - \tfrac{\nu}{2}\|x-y\|^2 \leqslant r(x), \qquad \forall x, y \in \operatorname{dom} r. \tag{4.2}$$

Here, we say that r satisfies (4.2) with parameter ν with respect to $\|\cdot\|$. In many cases, it is

possible that a convex function r is a combination of Lipschitz smooth and nonsmooth functions. Let $\omega : X \to \mathbb{R}$ be continuously differentiable with L_ω Lipschitz gradient and 1-strongly convex with respect to $\|\cdot\|$. We define the prox-function associated with $\omega(\cdot)$ as

$$W(y,x) := \omega(y) - \omega(x) - \langle \nabla\omega(x), y - x\rangle, \qquad \forall x, y \in X. \tag{4.3}$$

Based on the smoothness and strong convexity of $\omega(x)$, we have the following relation

$$W(y,x) \leqslant \tfrac{L_\omega}{2}\|x-y\|^2 \leqslant L_\omega W(x,y), \qquad \forall x, y \in X. \tag{4.4}$$

Moreover, we say that a function $r(\cdot)$ is β-strongly convex with respect to $W(\cdot,\cdot)$ if

$$r(x) \geqslant r(y) + \langle \nabla r(y), x - y\rangle + \beta W(x,y), \qquad \forall x, y \in X. \tag{4.5}$$

For any convex function h, we denote the subdifferential as ∂h which is defined as follows: at a point x in the relative interior of X, ∂h is comprised of all subgradients h' of h at x which are in the linear span of $X - X$. For a point $x \in X \backslash \operatorname{rint} X$, the set $\partial h(x)$ consists of all vectors h', if any, such that there exists $x_i \in \operatorname{rint} X$ and $h'_i \in \partial h(x_i), i = 1, 2, \ldots$, with $x = \lim\limits_{i\to\infty} x_i,\ h' = \lim\limits_{i\to\infty} h'_i$. With this definition, it is well-known that, if a convex function $h : X \to \mathbb{R}$ is Lipschitz continuous, with constant $\mathcal{M}$, with respect to a norm $\|\cdot\|$, then the set $\partial h(x)$ is nonempty for any $x \in X$ and

$$h' \in \partial h(x) \Rightarrow |\langle h', d\rangle| \leqslant \mathcal{M}\|d\|, \forall d \in \operatorname{lin}(X - X),$$

which also implies

$$h' \in \partial h(x) \Rightarrow \|h'\|_* \leqslant \mathcal{M},$$

where $\|\cdot\|_*$ is the dual norm. See [11] for more details.

4.2 Proximal Point Methods for Nonconvex Function Constrained Problems

Our goal in this section is to extend the ConEx method for the nonconvex setting by developing a general proximal point method for nonconvex function constrained optimization. This proximal point method transforms the nonconvex function constrained problem (3.1) into a sequence of convex function constrained subproblems. In Section 4.2.1, we present an exact proximal point method which carries its name since we assume that convex subproblems are solved exactly. This method requires a weak assumption on constraint qualification. Section 4.2.2 discusses an inexact proximal point method where convex subproblems are solved inexactly using the ConEx method presented in Chapter 3. Convergence of this method requires a stronger but verifiable constraint qualification.

We first recall the assumptions mentioned briefly in Section 3.1 for the nonconvex case.

1. $f_i : X \to \mathbb{R}$ are nonconvex and Lipschitz-smooth functions satisfying the lower curvature condition in (4.2) with parameters $\mu_i,\ i = 0, \ldots, m$.

2. $\chi_0 : X \to \mathbb{R}$ is a proper convex lower semicontinuous function.

3. $\chi_i : X \to \mathbb{R},\ i = 1, \ldots, m$ are convex and continuous functions.

Let $x^* \in X$ be a the global optimal solution and $\psi_0^* = \psi_0(x^*)$ be optimal value of problem (3.1). Given the above assumptions and compactness of X, we have $\psi_0^* > -\infty$.

It should be noted, however, that solving nonconvex problem (3.1) to the optimality condition in Definition 3.3.1 is generally difficult. Due to the hardness of the problem, we focus on the necessary condition for guaranteeing local optimality. For this purpose, we need to generalize the *subdifferential* for the objective function ψ_0 and constraints ψ_i because they are possibly nonconvex and nonsmooth. Let $\partial\chi_0$ and $\partial\chi_i, i \in [m]$ be the subdifferentials of the convex functions χ_0 and

$\chi_i, i \in [m]$, respectively. We define

$$\begin{aligned}\partial\psi_0(x) &:= \{\nabla f_0(x)\} + \partial\chi_0(x)\\ \partial\psi_i(x) &:= \{\nabla f_i(x)\} + \partial\chi_i(x), \qquad i \in [m].\end{aligned}$$

Note that $\partial\psi_i = \{\nabla f_i\}$ when ψ is a "purely" differentiable nonconvex function f_i and $\partial\psi_i = \partial\chi_i$ when ψ_i is a nonsmooth convex function χ_i.

Using these objects, we can define a Karush-Kuhn-Tucker (KKT) condition for this class of nonsmooth nonconvex problem (3.1) as follows.

Definition 4.2.1 *We say that $x^* \in X$ is a critical KKT point of* (3.1) *if $\psi_i(x^*) \leqslant 0$ and $\exists\ y^* = [y^{*(1)}, \ldots, y^{*(m)}]^T \geqslant \mathbf{0}$ s.t.*

$$\begin{aligned} y^{*(i)}\psi_i(x^*) &= 0, \qquad i \in [m],\\ d\big(\partial\psi_0(x^*) + \textstyle\sum_{i=1}^m y^{*(i)}\partial\psi_i(x^*) + N_X(x^*), \mathbf{0}\big) &= 0.\end{aligned} \tag{4.6}$$

The parameters $\{y^{*(i)}\}_{i\in[m]}$ are called *Lagrange multipliers*. For brevity, we use the notation y^* and $[y^{*(1)}, \ldots, y^{*(m)}]^T$ interchangeably.

It is well-known that for solving nonlinear optimization problems where functions ψ_0 and ψ_i's are continuously differentiable, the KKT condition is necessary for achieving optimality under the classical Mangasarian-Fromovitz constraint qualification (MFCQ, see [72]). Using the subdifferential $\partial\psi_0$ and $\partial\psi_i$ defined above, we will show that the KKT condition in (4.6) is a first-order necessary optimality condition for the composite nonconvex optimization problem in (3.1) under the following MFCQ type assumption.

Assumption 4.2.1 (MFCQ) *There exists a direction $z \in -N_X^*(x^*)$ such that*

$$\max_{v\in\partial\psi_i(x^*)} v^T z < 0, \qquad i \in \mathcal{A}(x^*), \tag{4.7}$$

where $\mathcal{A}(x^*)$ denotes the indicator set of all active constraints.

Proposition 4.2.1 below gives a necessary condition for a point to be a locally optimal solution of the problem (3.1) and its proof is given in Appendix 4.3.1.

Proposition 4.2.1 *Let x^* be a local optimal solution of the problem* (3.1)*. If x^* satisfies Assumption 4.2.1, then there exists $y^{*(i)} \geqslant 0,\ i \in [m]$ such that 4.6 holds.*

Due to the hardness of exactly computing even the local optimal solution for the nonconvex function constrained problem, it is natural to seek an approximate KKT point defined as follows.

Definition 4.2.2 *We say that a point $\widehat{x} \in X$ is an (ϵ, δ)-KKT point for the problem* (3.1) *if there exists (x, y) such that $\phi(x) \leqslant \mathbf{0}, y \geqslant \mathbf{0}$ and*

$$\begin{aligned} \textstyle\sum_{i=1}^m |y^{(i)}\psi_i(x)| &\leqslant \epsilon, \\ \left[d\big(\partial\psi_0(x) + \textstyle\sum_{i=1}^m y^{(i)}\partial\psi_i(x) + N_X(x), \mathbf{0}\big)\right]^2 &\leqslant \epsilon, \\ \|x - \widehat{x}\|^2 &\leqslant \delta. \end{aligned} \tag{4.8}$$

Similarly a stochastic (ε, δ)-KKT point generated by stochastic algorithms can be defined as a point $\widehat{x} \in X$ such that (4.8) is satisfied under expectation with respect to the random variables involved in these methods. Note that if $\delta = 0$ then $\widehat{x}$ coincides with x. In this case, we call $\widehat{x}$ as an ε-KKT point by dropping δ in the notation. Clearly a 0-KKT point satisfies the KKT condition (4.6) exactly since both $\varepsilon = \delta = 0$. The parameter δ in the approximation criterion (4.8) is introduced to discuss the convergence rate of our algorithm when the constrained convex subproblems in each iteration are solved inexactly. Termination criterion with $\delta > 0$ has been used in [60, 27] when solving the subproblems of the proximal point methods inexactly. However, under exact oracle for the subproblems, there is no need to use δ and in this case, we work with the stronger ε-KKT approximation criterion.

4.2.1 Exact proximal point method

The main idea of the proximal point method (see Algorithm 2) is to translate the nonconvex problem into a sequence of convex subproblems by adding strongly convex terms to the objective and

to the constraints. Specifically, each step of the proximal point algorithm involves a convex subproblem (4.9). It can be observed that, by adding a strongly convex proximal term, $\psi_0(x; x_{k-1})$ is μ_0-strongly convex and $\psi_i(x; x_{k-1})$ is μ_i-strongly convex with respect to $W(\cdot, \cdot)$. Hence, each subproblem will have a unique global optimal solution. Our main goal in this subsection is to analyze

Algorithm 2 Exact Constrained Proximal Point Algorithm

Input: Input x_0

1: **for** $k = 1, \ldots, K$ **do**

2: Set
$$\begin{aligned} \psi_0(x; x_{k-1}) &:= \psi_0(x) + 2\mu_0 W(x, x_{k-1}), \\ \psi_i(x; x_{k-1}) &:= \psi_i(x) + 2\mu_i W(x, x_{k-1}), \qquad i \in [m]. \end{aligned}$$

3: Obtain
$$\begin{aligned} x_k = \underset{x \in X}{\operatorname{argmin}} \quad & \psi_0(x; x_{k-1}) \\ \text{s.t.} \quad & \psi_i(x; x_{k-1}) \leqslant 0, \qquad i \in [m]. \end{aligned} \tag{4.9}$$

4: **If** $x_{k-1} = x_k$ then **return** x_k.

5: **end for**

6: **return** x_K

the convergence behavior of Algorithm 2. We will first describe some basic properties of Algorithm 2, e.g., monotonic nonincreasing objective values, square summability of distances between the consecutive iterates, etc. Moreover, by properly imposing constraint qualification assumptions, we will establish the asymptotic convergence and rate of convergence of this method to compute an approximate KKT point of problem (3.1).

Theorem 4.2.2 describes some basic properties of Algorithm 2, namely, the square summability of $x_{k-1} - x_k$ and sufficient descent property.

Theorem 4.2.2 *Assume that x_0 is feasible for* (3.1) *in Algorithm 2. Then*

a) Either the algorithm terminates at $x_1 = x_0$ or all the generated points $x_1, x_2, ..., x_k ...$ are strictly feasible for problem (3.1)*, and satisfy*

$$\textstyle\sum_{k=1}^{K} \|x_{k-1} - x_k\|^2 \leqslant \frac{2}{3\mu_0}[\psi_0(x_0) - \psi_0(x_K)], \tag{4.10}$$

$\{\psi_0(x_k)\}$ *is monotonically decreasing.*

b) Either there exists a $\widehat{k}$ such that $x_{\widehat{k}} = x_{\widehat{k}-1}$, and then the algorithm terminates, or $\{\psi_0(x_k)\}$ is strictly decreasing and has a limit point $\widetilde{\psi}_0 > -\infty$. In that case we have

$$\lim_{k\to+\infty} \|x_k - x_{k-1}\| = 0.$$

Proof. We first show part a). Note that x_0 is a feasible solution of subproblem (4.9) for $k = 1$. By definition, the optimal solution of this problem is x_1. If $x_1 = x_0$ then we have nothing to prove. We assume that $x_1 \neq x_0$. Since $\psi_i(x_1; x_0) \leqslant 0$ for all $i \in [m]$. Hence, we have $\psi_i(x_1) < 0$ for all $i \in [m]$ implying that x_1 is strictly feasible. Moreover, by continuity of ψ_i, we have that $\mathrm{int}(\{\psi \leqslant 0\}) \neq \varnothing$.

We prove the rest of the claim by induction. Assume that our claim holds for x_{k-1}, i.e., $\psi_i(x_{k-1}) < 0$, then x_{k-1} is strictly feasible for the k-th subproblem (4.9) with objective $\psi_0(\cdot; x_{k-1})$ and constraints $\psi(\cdot; x_{k-1})$. If $x_k = x_{k-1}$, the claim holds by the induction assumption. Otherwise, by the feasibility of x_k for (4.9), we have $\psi_i(x_k) < \psi_i(x_k; x_{k-1}) \leqslant 0$ for all $i \in [m]$.

Due to the optimality of x_k for solving subproblem (4.9) and noting the strong convexity of objective function $\psi_0(\cdot; x_{k-1})$, we have for all feasible x that $\psi_0(x; x_{k-1}) \geqslant \psi_0(x_k; x_{k-1}) + \mu_0 W(x, x_k)$. By inductive hypothesis, we have x_{k-1} is a feasible solution. Hence, taking $x = x_{k-1}$, and using strong convexity of the distance generating function $\omega(x)$ of $W(\cdot, \cdot)$, we have

$$\|x_{k-1} - x_k\|^2 \leqslant \tfrac{2}{3\mu_0}[\psi_0(x_{k-1}) - \psi_0(x_k)]. \tag{4.11}$$

Summing up (4.11) for $k = 1, 2, 3, ...K$ yields the result in part a).

To show part b), we observe from (4.11) that $\{\psi_0(x_k)\}$ is a nonincreasing sequence. Moreover, we have strict monotonicity if $x_k \neq x_{k-1}$ for all k. In that case we conclude that $\lim_{k\to+\infty} \psi_0(x_k) = \widetilde{\psi}_0$ for some $\widetilde{\psi}_0 \geqslant \psi_0^*$ and $\lim_{k\to+\infty} \|x_k - x_{k-1}\| = 0$. □

Strict feasibility is a common assumption to show the existence of Lagrange multipliers for convex programming. Henceforth, we will assume that the initial point x_0 is a strict feasible

solution for the problem (3.1) throughout this section. Then, in view of Theorem 4.2.2, we note that there exists a strict feasible solution for the subproblem (4.9) for all $k \geqslant 1$. Therefore, there exists a KKT point (x_k, y_k) based on Slater constraint qualification. The following lemma characterizes an important property of (x_k, y_k) for such convex nonlinear problems.

Lemma 4.2.3 *Let (x_k, y_k) be a KKT point of the subproblem* (4.9)*. Then*

$$\psi_0(x; x_{k-1}) - \psi_0(x_k; x_{k-1}) + \langle y_k, \psi(x; x_{k-1})\rangle \geqslant \left(\mu_0 + \mu^T y_k\right) W(x, x_k), \quad x \in X. \tag{4.12}$$

Proof. Let $\psi_0'(x_k) \in \partial\psi_0(x^*)$, $\psi_i'(x^*) \in \partial\psi_i(x^*)$ and $z^* \in N_X(x^*)$ be the subgradients satisfying the condition (4.6). According to the strong convexity of $\psi_0(\cdot; x_{k-1})$, $\psi_i(\cdot; x_{k-1})$, and the fact that $y_k \geqslant 0$, we have

$$\begin{aligned}
\psi_0(x; x_{k-1}) + \langle y_k, \psi(x; x_{k-1})\rangle &\geqslant \psi_0(x_k; x_{k-1}) + \langle \psi_0'(x_k; x_{k-1}), (x - x_k)\rangle + \mu_0 W(x, x_k) \\
&\quad + \langle y_k, \psi(x_k; x_{k-1})\rangle + \langle \textstyle\sum_{i=1}^m y_k^{(i)} \psi_i'(x_k; x_{k-1}), x - x_k\rangle + (\mu^T y_k) W(x, x_k) \\
&= \psi_0(x_k; x_{k-1}) + \langle \psi_0'(x_k; x_{k-1}) + \textstyle\sum_{i=1}^m y_k^{(i)} \psi_i'(x_k; x_{k-1}), x - x_k\rangle \\
&\quad + \left(\mu_0 + \mu^T y_k\right) W(x, x_k),
\end{aligned}$$

where the last equality follows from the complementary slackness part of KKT condition. Moreover, for all $x \in X$, we have

$$\langle \psi_0'(x_k; x_{k-1}) + \textstyle\sum_i y_k^{(i)} \psi_i'(x_k; x_{k-1}), x - x_k\rangle \geqslant 0,$$

where the inequality follows from the definition of normal cone. Putting the above two inequalities together, we arrive at relation (4.12). □

Note that even though Lemma 4.2.3 is stated for subproblem (4.9), it is applicable for any strongly convex function constrained problem. Using the above lemma and Theorem 4.2.2, we can show a bound on the norm of dual variables.

Proposition 4.2.4 *Assume that x_0 is strictly feasible for* (3.1) *in Algorithm 2. Then for all $k \geqslant 1$, there exists $y_k = [y_k^{(1)}, \ldots, y_k^{(m)}]^T$ such that $y_k \geqslant \mathbf{0}$, and*

$$\begin{aligned} y_k^{(i)} \psi_i(x_k; x_{k-1}) &= 0, \qquad i = 1, \ldots, m, \\ \partial\psi_0(x_k; x_{k-1}) + \textstyle\sum_{i\in[m]} y_k^{(i)} \partial\psi_i(x_k; x_{k-1}) + N_X(x_k) &\ni \mathbf{0}. \end{aligned} \tag{4.13}$$

and we have the following bound on y_k:

$$\|y_k\|_1 \leqslant \tfrac{\psi_0(x_{k-1}) - \psi_0(x_k)}{\min_{1\leqslant i\leqslant m}\{-\psi_i(x_{k-1})\}}, \quad k = 1, 2, 3, \ldots \tag{4.14}$$

Proof. Strict feasibility of x_0 along with Part (a) of Theorem 4.2.2 imply that each subproblem (4.9) in Algorithm 2 satisfies Slater constraint qualification for all $k \geqslant 1$. Hence, (4.13) follows from KKT necessary condition with Slater constraint qualification. In particular, first relation in (4.13) is a direct application of KKT complementary slackness and second relation is an application of KKT stationarity. Similarly, applying Lemma 4.2.3 and placing $x = x_{k-1}$ in (4.12) yields

$$\begin{aligned} \psi_0(x_{k-1}) - \psi_0(x_k) &\geqslant (\mu_0 + \mu^T y_k) W(x_{k-1}, x_k) + 2\mu_0 W(x_k, x_{k-1}) - \textstyle\sum_{i=1}^m y_k^{(i)} \psi_i(x_{k-1}) \\ &\geqslant \|y_k^{(i)}\|_1 \min_{1\leqslant i\leqslant m}\{-\psi_i(x_{k-1})\}. \end{aligned}$$

Thus relation (4.14) immediately follows. □

In view of Proposition 4.2.4, strict feasibility assumption implies a bound on y_k for each $k \geqslant 1$. As a special case, if $x_k = x_{k-1}$ for some $k > 1$, then the critical KKT point is in the interior of the inequality constraints and consequently, we have $y_k = \mathbf{0}$. Conceptually, we hope that the bound on the sequence $\{y_k\}$ and proximity of consecutive elements of the sequence $\{x_k\}$ leads to convergence to the KKT condition of the problem (3.1). However, Proposition 4.2.4 does not precisely describe the limiting behavior of the dual sequence, $\{y_k\}$. For instance, it does not preclude the case that the limit of the sequence $\|y_k\|_1$ tends to infinity, which is possible when x_k

converges to boundary points. In the latter case, mere existence of the optimal dual multiplier y_k of the subproblem does not necessarily implies convergence to a solution satisfying KKT condition of the problem (3.1). We indeed need to analyze under what conditions one can definitively say that for the entire sequence of subproblems generated by Algorithm 2, the optimal dual variables remain bounded. In what follows, we describe two sufficient conditions under which convergence to the KKT solutions can be established. We show that the assumptions are relatively weak in the sense that they are satisfied some variants of MFCQ which is a classical constraint qualification for function constrained problems.

Assumption 4.2.2 (Subsequence boundedness) *Given the sequence of primal variables $\{x_k\}_{k=1}^{\infty}$, one limit point x^*, and the sequence of optimal dual variables $\{y_k\}_{k=1}^{\infty}$, if $\{x_{i_k}\}$ is a subsequence convergent to x^*, then the subsequence $\{y_{i_k}\}$ is bounded.*

The following lemma shows that MFCQ implies the subsequence boundedness condition.

Lemma 4.2.5 *In Algorithm 2, let x^* be a limit point of the sequence $\{x_k\}$. Assume that there exists some $z \in -N_X^*(x^*)$ such that Assumption 4.2.1 is satisfied, then Assumption 4.2.2 is satisfied.*

Proof. We prove by contradiction, that the dual variable associated with the convergent subsequence is bounded.

Let $x^* \in X$ be a limit point of the sequence $\{x_k\}$. Passing to a subsequence if necessary, we have $\lim_{k\to\infty} x_k = x^*$. For the sake of contradiction, assume that $\{y_k\}$ is not bounded. Then there exists a subsequence $\{j_k\}$ such that $\lim_{k\to\infty}\|y_{j_k}\|_1 = \infty$. Due to the optimality of x_{j_k}, we have

$$\psi_0(x_{j_k}) + y_{j_k}^T\psi(x_{j_k}) \leqslant \psi_0(x) + y_{j_k}^T\psi(x) + 2(\mu_0 + \mu^T y_{j_k})[W(x, x_{j_k-1}) - W(x_{j_k}, x_{j_k-1})], \quad \forall x \in X. \tag{4.15}$$

Let $v_{j_k} = y_{j_k}/\|y_{j_k}\|_1$, then $\|v_{j_k}\|_1 = 1$, hence $\{v_{j_k}\}$ must have a convergent subsequence. Without loss of generality, we assume $\lim_{k\to\infty} v_{j_k} = v^*$. Dividing both sides of (4.15) by $\|y_{j_k}\|_1$, taking $k \to \infty$ and using continuity of ψ, we have

$$v^{*T}\psi(x^*) = \lim_{k\to\infty} v^{*T}\psi(x_{j_k}) \leqslant v^{*T}\psi(x) + 2\mu^T v^* W(x, x^*), \quad \forall x \in X. \tag{4.16}$$

Given that x^* is optimal, the first-order necessary condition implies

$$d\big(\textstyle\sum_i \partial\psi_i(x^*)v^{*(i)} + N_X(x^*), \mathbf{0}\big) = 0. \tag{4.17}$$

Let $\mathcal{A}(x^*)$ be the set of active constraints at x^*. By this definition, for any $i \notin \mathcal{A}(x^*)$, we have $\psi_i(x^*) < 0$. Since ψ_i is continuous and $\|x_{j_k} - x_{j_{k-1}}\|^2$ converges to 0, there exists k_0 such that for all $k > k_0$, we have $\psi_i(x_{j_k}; x_{j_k-1}) < 0$. Hence, according to the KKT complementary slackness condition for the subproblem, $y_{j_k}^{(i)} = 0$ for $k > k_0$. Taking $k \to \infty$ we obtain $v^{*(i)} = 0$ for any $i \notin \mathcal{A}(x^*)$. So we can rewrite the equation (4.17) as

$$d\Big(\textstyle\sum_{i\in\mathcal{A}(x^*)} \partial\psi_i(x^*)v^{*(i)} + N_X(x^*), \mathbf{0}\Big) = 0.$$

Let $\psi_i'(x^*) \in \partial\psi_i(x^*),\ i \in [m]$, and $u \in N_X(x^*)$ be such that $u + \sum_{i=1}^m \psi_i'(x^*)v^{*(i)} = \mathbf{0}$. Then,

$$\begin{aligned} 0 = z^T u + \textstyle\sum_{i\in\mathcal{A}(x^*)} v^{*(i)} z^T \psi_i'(x^*) &\leqslant \textstyle\sum_{i\in\mathcal{A}(x^*)} v^{*(i)} z^T \psi_i'(x^*) \\ &\leqslant \textstyle\sum_{i\in\mathcal{A}(x^*)} v^{*(i)} \max_{v\in\partial\psi_i(x^*)} z^T v < 0, \end{aligned}$$

where the first inequality follows since $z \in -N_X^*(x^*)$ and $u \in N_X(x^*)$ hence $z^T u \leqslant 0$, the second inequality follows due to the fact that $v^{*(i)} \geqslant 0$ and $\psi_i'(x^*) \in \partial\psi_i(x^*)$ and the last strict inequality follows due to Assumption 4.2.1 and $v^{*(i)} > 0$ for at least one $i \in \mathcal{A}(x^*)$. Hence, we obtain a contradiction and conclude that $\{y_{j_k}\}$ is a bounded sequence and finish the proof. □

We are now ready to state our first general convergence result for Algorithm 2.

Theorem 4.2.6 *Let x^* be a limit point of Algorithm 2. If Assumption 4.2.2 holds, then there exists a vector $y^* \geqslant 0$ such that the KKT conditions in* (4.6) *are satisfied.*

Proof. From the KKT condition for the k-th subproblem and noting that

$$\partial\psi_0(\cdot; x_{k-1}) = \partial\psi_0(\cdot) + 2\mu_0(\nabla\omega(\cdot) - \nabla\omega(x_{k-1})),$$
$$\partial\psi_i(\cdot; x_{k-1}) = \partial\psi_i(\cdot) + 2\mu_i(\nabla\omega(\cdot) - \nabla\omega(x_{k-1})),$$

we have

$$y_k^{(i)}\psi_i(x_k) = -2y_k^{(i)}\mu_i W(x_k, x_{k-1}), \qquad i = 1, \ldots, m, \tag{4.18}$$

and

$$\begin{aligned} d\big(\partial\psi_0(x_k) + \textstyle\sum_{i=1}^m y_k^{(i)}\partial\psi_i(x_k) + N_X(x_k), \mathbf{0}\big) &\leqslant 2\big(\mu_0 + \mu^T y_k\big)\,\|\nabla\omega(x_k) - \nabla\omega(x_{k-1})\| \\ &\leqslant 2\sqrt{2}L_\omega(\mu_0 + \|\mu\|_\infty\|y_k\|_1)\sqrt{W(x_{k-1}, x_k)}. \end{aligned} \tag{4.19}$$

Applying Lemma 4.2.3 with $x = x_{k-1}$, we have

$$\psi_0(x_{k-1}) - \psi_0(x_k) \geqslant 2\mu_0 W(x_k, x_{k-1}) + (\mu_0 + \mu^T y_k)\, W(x_{k-1}, x_k). \tag{4.20}$$

Together with (4.18) we obtain

$$\begin{aligned} \textstyle\sum_{i=1}^m |y_k^{(i)}\psi_i(x_k)| = 2\big(\mu^T y_k\big)W(x_k, x_{k-1}) &\leqslant 2L_\omega\big(\mu^T y_k\big)W(x_{k-1}, x_k) \\ &\leqslant 2L_\omega[\psi_0(x_{k-1}) - \psi_0(x_k)], \end{aligned} \tag{4.21}$$

where the first inequality follows from (4.4).

In view of the convergence of $\{\psi_0(x_k)\}$ according to Theorem 4.2.2, we have

$$\lim_{k\to\infty} y_k^{(i)}\psi_i(x_k) = 0, \qquad i = 1, 2, ..., m.$$

Let $\{x_{j_k}\}$ be a convergent subsequence to x^*. Based on Assumption 4.2.2, $\|y_{j_k}\|$ is bounded above. Passing to a subsequence if necessary, we have $\lim_{k\to\infty} y_{j_k} = y^*$. Then $y^* \geqslant \mathbf{0}, \psi(x^*) \leqslant \mathbf{0}$

and

$$y^{*(i)}\psi_i(x^*) = 0, \qquad i = 1, \ldots, m. \tag{4.22}$$

Moreover, using part two of Theorem 4.2.2 we have $\lim_{k\to\infty} \psi_0(x_{j_k}) = \widetilde{\psi}_0 > -\infty$. We will show $\psi_0(x^*) = \widetilde{\psi}_0$. First, due to lower semicontinuity of ψ_0, we have $\psi_0(x^*) \leqslant \widetilde{\psi}_0$. Next, taking $k \to \infty$ in (4.15) in Lemma 4.2.5, noting the definition of $\widetilde{\psi}_0$ and continuity of ψ, we have

$$\widetilde{\psi}_0 + y^{*T}\psi(x^*) \leqslant \psi_0(x) + y^{*T}\psi(x) + 2(\mu_0 + \mu^T y^*)W(x, x^*), \quad \forall x \in X. \tag{4.23}$$

Plugging the value $x = x^*$ in the above relation, we have $\psi_0(x^*) \geqslant \widetilde{\psi}_0$. Consequently, we have $\psi_0(x^*) = \widetilde{\psi}_0$. Replacing $\widetilde{\psi}_0$ by $\psi_0(x^*)$ in the condition (4.23), the optimality of x^* implies

$$d\big(\partial\psi_0(x^*) + \textstyle\sum_{i=1}^m y^{*(i)}\partial\psi_i(x^*) + N_X(x^*), \mathbf{0}\big) = 0. \tag{4.24}$$

Here note that we dropped the term, $\nabla\omega(\cdot) - \nabla\omega(x^*)$, which evaluates to $\mathbf{0}$ at x^*. From equations (4.22), (4.24) and the assertion that $y^* \geqslant \mathbf{0}$ and $\psi(x^*) \leqslant \mathbf{0}$, we conclude that (x^*, y^*) is a KKT point of problem (3.1). □

Our goal in the remaining part of this subsection is to develop the iteration complexity, i.e., a bound on the number of iterations performed by Algorithm 2 in order to obtain an ε-KKT point, as specified in Definition 4.2.2. To achieve this goal, we require a stronger assumption of uniform bounded dual sequence.

Assumption 4.2.3 (Uniform boundedness) *Given the sequence of optimal dual variables $\{y_k\}$ of subproblem* (4.9)*, the whole sequence $\{y_k\}$ is bounded:*

$$\exists B > 0 \quad \textit{s.t.} \quad \|y_k\|_1 \leqslant B, \qquad k = 1, 2, ..., \tag{4.25}$$

In the following lemma, we show that uniform boundedness of dual variables can be guaranteed

under some mild conditions.

Lemma 4.2.7 *If Assumption 4.2.2 holds for every limit point x^* of Algorithm 2 , then Assumption 4.2.3 also holds.*

Proof. The boundedness of y_k can be proved by contradiction. Suppose that there exists an unbounded subsequence $\{y_{i_k}\}$ such that $\lim_{k\to\infty}\|y_{i_k}\|_1 = \infty$. Since X is a compact set and $\{x_{i_k}\}$ is a bounded sequence, there exists a convergent subsequence $\{j_k\} \subseteq \{i_k\}$: $\lim_{k\to\infty} x_{j_k} = x^*$. However, $\{y_{j_k}\}$ is bounded according to Assumption 4.2.2. Hence we have a contradiction. □

Below, we state an immediate corollary of Lemma 4.2.5 and Lemma 4.2.7 which gives uniform bounds on the the sequence $\|y_k\|_1$ using a stronger version of MFCQ.

Corollary 4.2.8 *Suppose $z \in -N_X^*(x^*)$ satisfying* (4.2.1) *exists for every limit point x^* of Algorithm 2 then Assumption 4.2.3 holds.*

In the corollary above, we used condition (4.2.1) for every limit point, x^*, of Algorithm 2 in order to show that Assumption 4.2.3 holds. However, it is difficult to verify whether this condition is satisfied. Alternatively, we provide another verifiable sufficient condition that ensures uniform boundedness assumption.

Lemma 4.2.9 *Let $D_X := \max_{x,y\in X}\sqrt{2W(x,y)}$. Suppose there exists $\bar{x} \in X$ such that*

$$\psi_i(\bar{x}) \leqslant -2\mu_i D_X^2, \qquad i = 1, \ldots, m. \tag{4.26}$$

Then Assumption 4.2.3 holds, and specifically, we have the following uniform bound:

$$\|y_k\|_1 \leqslant B := \tfrac{\psi_0(\bar{x})-\psi_0^*+\mu_0 D_X^2}{\mu_{\min} D_X^2}, \qquad k = 1, 2, 3, ..., \tag{4.27}$$

where $\mu_{\min} = \min_{1\leqslant i\leqslant m}\mu_i$.

Proof. Based on (4.26), for subproblem 4.9, we have

$$\psi_i(\bar{x}, x_{k-1}) \leqslant -2\mu_i D_X^2 + 2\mu_i W(\bar{x}, x_{k-1}) \leqslant -\mu_i D_X^2 < 0.$$

Then the existence of the KKT point (x_k, y_k) follows from the Slater condition. Moreover, using $x = \bar{x}$ in Lemma 4.2.3, and noting that $y_k \geqslant \mathbf{0}$, one has

$$\psi_0(\bar{x}) + 2\mu_0 W(\bar{x}, x_{k-1}) - \psi_0(x_k) - 2\mu_0 W(x_k, x_{k-1}) \geqslant \langle y_k, -\psi(\bar{x}, x_{k-1})\rangle.$$

Combining the above two inequalities together, we successively deduce

$$\begin{aligned}
\mu_{\min}\|y_k\|_1 D_X^2 &\leqslant (\mu^T y_k) D_X^2 \\
&\leqslant -\textstyle\sum_i y_k^{(i)} \psi_i(\bar{x}, x_{k-1}) \\
&\leqslant \psi_0(\bar{x}) - \psi_0(x_k) + \mu_0 D_X^2, \quad k = 1, 2, \ldots, K.
\end{aligned}$$

Finally, since the feasible region of the subproblem 4.9 is smaller than that of Problem 3.1, we have $\psi_0(x_k) \geqslant \psi_0^*$. The result immediately follows. □

Note that (4.26) is a local and a verifiable condition and it provides a computable uniform bound B, as in accordance with the result of Lemma 4.2.9. While it appears that (4.26) is quite distinct from Assumption 4.2.1, we would like to point out certain similarities between these two conditions. To understand this connection better, let us assume that ψ_i is smooth function. Then for all $x \in X$, we have

$$\begin{aligned}
\psi_i(\bar{x}) &\geqslant \psi_i(x) + \langle \nabla\psi_i(x), \bar{x} - x\rangle - \tfrac{\mu_i}{2}\|\bar{x} - x\|^2 \\
\Rightarrow \langle \nabla\psi_i(x), x - \bar{x}\rangle &\geqslant \psi_i(x) - \psi_i(\bar{x}) - \tfrac{\mu_i}{2}\|\bar{x} - x\|^2,
\end{aligned}$$

which implies that

$$\langle \nabla \psi_i(x), x - \bar{x} \rangle \geqslant 0, \qquad \forall x \in X \cap \{\psi_i \geqslant -\frac{3}{2}\mu_i D_X^2\}. \tag{4.28}$$

Recall that the existence of a Minty solution, $\bar{x}$, for variational inequality problem on mapping $\nabla \psi_i$, is the following condition

$$\langle \nabla \psi_i(x), x - \bar{x} \rangle \geqslant 0, \qquad \forall x \in X, \tag{4.29}$$

which is a stronger condition than (4.28). Hence, ψ satisfying (4.26) is not necessarily quasi-convex. However, existence of Minty solution, $\bar{x}$, gives an 'almost' sufficient condition for ensuring Assumption 4.2.1 in the following way. Set $x = x^*$ in (4.29). Then we obtain that $z = \bar{x} - x^*$ satisfies Assumption 4.2.1 with strict inequality replaced by nonstrict inequality. Since there is no implication from (4.28) to (4.29) (in fact, the implication is in the opposite direction), so a direct comparison for the weaker among the two condition (4.26) and Assumption 4.2.1, can not be made as such.

Having provided with two sufficient conditions for the uniform boundedness assumption, we now present the main complexity result of Algorithm 2 in the following theorem.

Theorem 4.2.10 *If the dual sequence $\{y_k\}$ is bounded, i.e., Assumption 4.2.3 holds such that $\|y_k\|_1 \leqslant B$, then for $\widehat{k} = \operatorname{argmin}_{1 \leqslant k \leqslant K} \psi_0(x_{k-1}) - \psi_0(x_k)$, $x_{\widehat{k}}$ is an ε_K-KKT point with*

$$\varepsilon_K = \max\{2L_\omega, 8L_\omega^2(\mu_0 + \|\mu\|_\infty B)\} \, \tfrac{[\psi_0(x_0) - \psi_0^*]}{K}.$$

Proof. We derive the complexity to compute an approximate KKT point. By definition of $\widehat{k}$,

$$K[\psi_0(x_{\widehat{k}-1}) - \psi_0(x_{\widehat{k}})] \leqslant \textstyle\sum_{k=1}^K [\psi_0(x_{k-1}) - \psi_0(x_k)] \leqslant \psi_0(x_0) - \psi_0^*. \tag{4.30}$$

Putting together (4.30), the relation (5.13), (4.20) and (4.44) we conclude that

$$\textstyle\sum_{i=1}^{m}|y_{\hat{k}}^{(i)}\psi_i(x_{\hat{k}})| \leqslant \frac{2L_\omega[\psi_0(x_0)-\psi_0^*]}{K},$$

and

$$\begin{aligned} d\big(\partial\psi_0(x_{\hat{k}}) + \textstyle\sum_{i=1}^{m} y_{\hat{k}}^{(i)}\partial\psi_i(x_{\hat{k}}) + N_X(x_{\hat{k}}), \mathbf{0}\big)^2 &\leqslant 4\big(\mu_0 + (\mu^T y_{\hat{k}})\big)^2 \|\nabla\omega(x_{\hat{k}}) - \nabla\omega(x_{\hat{k}-1})\|^2 \\ &\leqslant 8L_\omega^2\big(\mu_0 + (\mu^T y_{\hat{k}})\big)^2 W(x_{\hat{k}-1}, x_{\hat{k}}) \\ &\leqslant 8L_\omega^2\big(\mu_0 + (\mu^T y_{\hat{k}})\big)\big[\psi_0(x_{\hat{k}-1}) - \psi_0(x_{\hat{k}})\big] \\ &\leqslant \tfrac{8L_\omega^2(\mu_0+\|\mu\|_\infty B)[\psi_0(x_0)-\psi_0^*]}{K}. \end{aligned} \tag{4.31}$$

Moreover, due to Part (a) of Theorem 4.2.2, we have $\psi(x_{\hat{k}}) \leqslant \mathbf{0}$ and due to Proposition 4.2.4, we have $y_{\hat{k}} \geqslant \mathbf{0}$. Hence we conclude the proof. □

In view of Theorem 4.2.10, the exact proximal point method finds an ε-KKT point. in $O(1/\varepsilon)$ iterations.

Remark 4.2.11 *Note that all the results in this section can be easily extended to the case when $\psi_i, i \in [m]$ are convex functions. In that case, we can replace $\mu_i = 0$ for all $i \in [m]$. This changes* (4.9) *of Algorithm 2 to*

$$\begin{aligned} x_k = \underset{x\in X}{\operatorname{argmin}} \quad & \psi_0(x; x_{k-1}) \\ \textit{s.t.} \quad & \psi_i(x) \leqslant 0, \qquad i \in [m]. \end{aligned} \tag{4.32}$$

Hence constraints are fixed for all iterations. For Algorithm 2 with (4.9) *replaced by* (4.32)*, we can easily obtain asymptotic convergence result of Theorem 4.2.6 for limits point x^* satisfying Assumption 4.2.1 with almost the same proof except replace μ_i by 0 for all $i \in [m]$ and $\psi(x; x_{k-1}) = \psi(x)$ for all $k \geqslant 1$. Under Assumption 4.2.1 for every limit point of $\{x_k\}$, we obtain rate of convergence result similar to Theorem 4.2.10 with almost the same proof and similar replacements.*

It should be noted that we need to assume access to an oracle that solves the convex subproblem (4.9) exactly in Algorithm 2. Such a problem can be efficiently solved by polynomial time algorithms, e.g., by the ellipsoid method and interior point methods, if the problem dimension is relatively small to medium. However, there exist scenarios where exact solutions are difficult to attain, e.g., when the objective or constraints are expectation of stochastic functions. Hence we turn our attention to an inexact proximal point algorithm which only requires approximate solution for the subproblem (4.9). We present details in the next subsection.

4.2.2 Inexact proximal point method

In this subsection, we propose an inexact variant of the proximal point method which solves the subproblem inexactly. To understand our motivation for the analysis of inexact proximal point method, consider the case when the objective function is given in the form of $f(x) = \mathbb{E}_\xi[F(x,\xi)]$, where $F(x,\xi)$ is a stochastic function on some random variable ξ and is possibly nonconvex with respect to the parameter x. Consequently, the objective function in the subproblem (4.9) is given by $\mathbb{E}_\xi[F(x,\xi)] + \mu_0\|x-\bar{x}\|^2$. As discussed in the previous section, stochastic optimization algorithms for solving this type of problem will exhibit a sublinear rate of convergence, making it difficult to attain high-precision solution.

Algorithm 3 Inexact Constrained Proximal Point Algorithm

1: Input x_0
2: **for** $k = 1, \ldots, K$ **do**
3: $\quad x_k \leftarrow$ a (stochastic) approximate solution of subproblem (4.9).
4: **end for**
5: Randomly choose $\widehat{k}$ from $\{1, 2, ..., K\}$.
6: **return** $x_{\widehat{k}}$.

To deal with this type of problem, we propose a (stochastic) inexact proximal point method as shown in Algorithm 3. The main difference between Algorithm 3 and Algorithm 2 is that the former permits approximate optimal solutions. To distinct exact and approximate solution, we denote exact solution as x_k^* and corresponding dual solution as y_k^* hereafter for this subsection. Since each subproblem (4.9) is solved inexactly, the sequence generated by Algorithm 3 can become infea-

sible with respect to the original problem. If x_{k-1} is infeasible with respect to (3.1), then we can not guarantee feasibility of the subproblem (4.9) in general. This also implies obtaining bounds on Lagrange multipliers is more challenging for inexact case. However, we show that if successive problems are solved accurately enough then we can obtain strict feasibility of the iterates and moreover, also show boundedness guarantees on $\|y_k\|_1$ as in the previous subsection.

Throughout the rest of this subsection, we assume that $\psi_0(\cdot; x_{k-1})$ is Lipschitz continuous with constant M_0, $\psi_i(\cdot; x_{k-1})$ is Lipschitz continuous with constant M_i, $i \in [m]$, and denote $M = [M_1, M_2, ..., M_m]^T$. Proposition 4.2.12 shows that the sequence $\{x_k\}$ is strictly feasible if the subproblem (4.9) is solved accurately enough.

Proposition 4.2.12 *Let $\{x_k\}$ be the sequence generated by Algorithm 3.*

a) For the subproblem (4.9)*, assume that $\psi(x_{k-1}) < 0$ and $x_k^* \neq x_{k-1}$. If x_k satisfies*

$$\sqrt{\tfrac{M_i}{\mu_i}\|x_k - x_k^*\|} + \|x_k - x_k^*\| < \|x_{k-1} - x_k^*\|, \qquad \textit{for all } i \in [m], \tag{4.33}$$

then x_k is a strictly feasible point for problem (3.1)*. If x_0 is strictly feasible, then the whole sequence $\{x_k\}$ is strictly feasible.*

b) Furthermore, if x_k satisfies:

$$\sqrt{\tfrac{2M_0}{\mu_0}\|x_k - x_k^*\|} + \|x_k - x_k^*\| \leqslant \|x_{k-1} - x_k^*\|, \tag{4.34}$$

then $\{\psi_0(x_k)\}$ is monotonically decreasing and converges to a limit point $\widetilde{\psi}_0$. Moreover we have

$$\lim_{k\to\infty} W(x_k, x_{k-1}),\ \lim_{k\to\infty} W(x_{k-1}, x_k^*) = 0. \tag{4.35}$$

Proof. Part a). Let us use $\varepsilon_k = \|x_k - x_k^*\|$ for brevity. From the definition of $\psi_i(x; x_{k-1})$ and feasibility of x_k^*, we have

$$\psi_i(x_k) + 2\mu_i W(x_k, x_{k-1}) = \psi_i(x_k; x_{k-1}) \leqslant \psi_i(x_k^*; x_{k-1}) + M_i\|x_k - x_k^*\| \leqslant M_i\|x_k - x_k^*\|,$$

where the first inequality follows from Lipschitz continuity of $\psi_i(x; x_{k-1})$. Using the triangle inequality, we have

$$\begin{aligned}\sqrt{2\mu_i W(x_k, x_{k-1})} \geqslant \sqrt{\mu_i}\|x_k - x_{k-1}\| &\geqslant \sqrt{\mu_i}(\|x_{k-1} - x_k^*\| - \|x_k - x_k^*\|) \\ &> \sqrt{M_i\|x_k - x_k^*\|}.\end{aligned}$$

Combining the above two results together, we have $\psi^{(i)}(x_k) < 0$.

Part b). We successively deduce

$$\begin{aligned}\psi_0(x_{k-1}) &= \psi_0(x_{k-1}; x_{k-1}) \\ &\geqslant \psi_0(x_k^*; x_{k-1}) - \langle y_k^*, \psi(x_{k-1}; x_{k-1})\rangle + (\mu_0 + \mu^T y_k^*)W(x_{k-1}, x_k^*) \\ &\geqslant \psi_0(x_k; x_{k-1}) - M_0\varepsilon_k + (\mu_0 + \mu^T y_k^*)W(x_{k-1}, x_k^*). \\ &= \psi_0(x_k) + 2\mu_0 W(x_k, x_{k-1}) - M_0\varepsilon_k + (\mu_0 + \mu^T y_k^*)W(x_{k-1}, x_k^*).\end{aligned}$$

Here the first inequality uses Lemma 4.2.3 with $x = x_{k-1}$ and replacing the saddle point (x_k, y_k) defined in Lemma 4.2.3 by (x_k^*, y_k^*). Together with (4.34), we deduce

$$\psi_0(x_k) + \mu_0 W(x_k, x_{k-1}) + (\mu_0 + \mu^T y_k^*)W(x_{k-1}, x_k^*) \leqslant \psi_0(x_{k-1}). \tag{4.36}$$

We immediately observe that $\psi_0(x_k)$ is decreasing. Since ψ_0 is bounded below, we have the convergence $\lim_k \psi_0(x_k) = \widetilde{\psi}_0$ for some $\widetilde{\psi}_0 > -\infty$. Summing up the above relation for $k = 1, 2, ...,$ we have

$$\sum_{k=1}^{\infty}[\mu_0 W(x_k, x_{k-1}) + (\mu_0 + \mu^T y_k^*)W(x_{k-1}, x_k^*)] \leqslant \psi_0(x_0) - \widetilde{\psi}_0 < +\infty. \tag{4.37}$$

Therefore, the last result immediately follows. □

The following lemma shows that MFCQ (Assumption 4.2.1) along with (4.33) and (4.34) is sufficient to guarantee dual boundedness assumptions for Algorithm 3.

Theorem 4.2.13 *In Algorithm 3, under all the assumptions of Proposition 4.2.12:*

a) If Assumption 4.2.1 holds at a limit point x^* of the sequence $\{x_k\}$, then Assumption 4.2.2 holds for sequence $\{x_k\}$ and $\{y_k^*\}$. Moreover, there exists a vector y^* the KKT conditions in (4.6) *are satisfied.*

b) If Assumption 4.2.1 holds at every limit point of $\{x_k\}$, then the whole sequence $\{y_k^*\}$ is uniformly bounded, i.e. Assumption 4.2.3 holds, i.e., $\|y_k\|_1 \leqslant B$ for some constant $B > 0$. Then after K iterations, there exists an $(\varepsilon_K, \bar{\varepsilon}_K)$-KKT point with $\varepsilon_K, \bar{\varepsilon}_K \in O(1/K)$.

Proof. Part a) Let $x^* \in X$ be a limit point of the sequence $\{x_k\}$ and let $\{x_{j_k}\}$ be a convergent subsequence to x^*. Denote $\{x_k^*\}$ the primal optimal solutions for the sequence of subproblems. Due to Proposition 4.2.12, $\lim_{k\to\infty} x_{j_k}^* = x^*$, hence x^* is also a limit point of sequence $\{x_k^*\}$. Using Lemma 4.2.5 we can show $y_{j_k}^*$ is bounded, hence concluding that Assumption 4.2.2 holds.

Applying Lemma 4.2.3 with $x = x_{k-1}$ and replacing (x_k, y_k) by (x_k^*, y_k^*), we have

$$\psi_0(x_{k-1}) - \psi_0(x_k^*) \geqslant 2\mu_0 W(x_k^*, x_{k-1}) + (\mu_0 + \mu^T y_k^*)\, W(x_{k-1}, x_k^*). \tag{4.38}$$

Together with (4.18) we obtain

$$\begin{aligned}
\textstyle\sum_{i=1}^m |y_k^{*(i)} \psi_i(x_k)| &= \textstyle\sum_{i=1}^m |y_k^{*(i)} \psi_i(x_k^*)| + \sum_{i=1}^m y_k^{*(i)} M_i \|x_k - x_k^*\| \\
&\leqslant 2\big(\mu^T y_k^*\big) W(x_k^*, x_{k-1}) + (M^T y_k^*)\, \|x_k - x_k^*\| \\
&\leqslant 2L_\omega\big(\mu^T y_k^*\big) W(x_{k-1}, x_k^*) + (M^T y_k^*)\, \|x_k - x_k^*\|.
\end{aligned} \tag{4.39}$$

Proposition 4.2.12 implies

$$\lim_{k\to\infty} y_k^{*(i)} \psi_i(x_k) = 0, \qquad i = 1, 2, ..., m.$$

Consider the limit point x^* of Algorithm 3, with $\{x_{j_k}\}$ being the subsequence convergent to x^*. Based on Assumption 4.2.2, $\{y_{j_k}^*\}$ is bounded. Passing to a subsequence if necessary, we have

$\lim_{k\to\infty} y^*_{j_k} = y^*$. Hence we have the complementary slackness:

$$y^{*(i)}\psi_i(x^*) = 0, \qquad i = 1, 2, ..., m.$$

The rest of the proof is slightly simplified from the proof of Theorem 4.2.6, since we assume that f is continuous. The KKT condition for the subproblem implies that

$$\psi_0(x^*_{j_k}) + {y^*_{j_k}}^T\psi(x^*_{j_k}) \leqslant \psi_0(x) + {y^*_{j_k}}^T\psi(x) + (2\mu_0 + \mu^T y^*_{j_k})W(x, x_{j_k-1}), \qquad \forall x \in X. \quad (4.40)$$

Taking $k \to \infty$ and using the continuity of ψ_0 and ψ, we have

$$\psi_0(x^*) + {y^*}^T\psi(x^*) \leqslant \psi_0(x) + {y^*}^T\psi(x), \qquad \forall x \in X. \quad (4.41)$$

Based on the optimality of x^* of minimizing the right hand side, we have $0 \in N_X(x^*) + \partial\psi_0(x^*) + \sum_{i\in[m]} y^{*(i)}\partial\psi_i(x^*)$. Hence (x^*, y^*) is a KKT point.

Part b). We show the boundedness of $\{y_k\}$ by contradiction. If there exists a subsequence $\{j_k\}$ such that $\lim_{k\to\infty} \|y^*_{j_k}\| = \infty$. Since $\{x_{j_k}\}$ is bounded, it has a limit point x^*. However, according to part a), $\|y^*_{j_k}\|$ is bounded, leading to a contradiction.

Furthermore, due to the KKT condition for (4.9), we have

$$d\big(\partial\psi_0(x^*_k; x_{k-1}) + \textstyle\sum_{i=1}^m y^{*(i)}_k \partial\psi_i(x^*_k; x_{k-1}) + N_X(x^*_k), \mathbf{0}\big) \ni 0.$$

Plugging the definition of $\partial\psi_0(; x_{k-1})$ and $\partial\psi_i(; x_{k-1}), i \in [m]$, into the above inequality yields

$$d\big(\partial\psi_0(x^*_k) + \textstyle\sum_{i=1}^m y^{*(i)}_k \partial\psi_i(x^*_k) + 2(\mu_0 + \mu^T y^*_k)(\nabla\omega(x^*_k) - \nabla\omega(x_{k-1})) + N_X(x^*_k), \mathbf{0}\big) = 0. \quad (4.42)$$

Applying inequality (4.36), we deduce

$$\begin{aligned} & d(\partial\psi_0(x_k^*) + \textstyle\sum_{i=1}^m y_k^{*(i)}\partial\psi_i(x_k^*) + N_X(x_k^*), \mathbf{0})^2 \\ & \leqslant (\mu_0 + \mu^T y_k^*)^2 \|\nabla\omega(x_k^*) - \nabla\omega(x_{k-1})\|^2 \\ & \leqslant 2L_\omega^2(\mu_0 + \mu_{\min}B)(\mu_0 + \mu^T y_k^*)W(x_{k-1}, x_k^*) \\ & \leqslant 2L_\omega^2(\mu_0 + \mu_{\min}B)[\psi_0(x_{k-1}) - \psi_0(x_k)]. \end{aligned} \tag{4.43}$$

In addition, by KKT condition we have

$$\begin{aligned} \textstyle\sum_{i=1}^m |y_k^{*(i)}\psi_i(x_k^*)| = 2\big(\mu^T y_k^*\big)W(x_k^*, x_{k-1}) & \leqslant 2L_\omega\big(\mu^T y_k^*\big)W(x_{k-1}, x_k^*) \\ & \leqslant 2L_\omega[\psi_0(x_{k-1}) - \psi_0(x_k)], \end{aligned} \tag{4.44}$$

where the last inequality is due to (4.36).

Furthermore, by the assumption of (4.34) and relation (4.36) we have $\|x_k - x_k^*\|^2 \leqslant \|x_{k-1} - x_k^*\|^2 \leqslant \frac{2}{\mu_0}[\psi_0(x_{k-1}) - \psi_0(x_k)]$. It can be seen that to obtain an approximate KKT solution with small error, it suffices to bound $\psi_0(x_{k-1}) - \psi_0(x_k)$. Since $\min_{1\leqslant k\leqslant K}[\psi_0(x_{k-1}) - \psi_0(x_k)] \leqslant \frac{1}{K}\sum_{k=1}^K[\psi_0(x_{k-1}) - \psi_0(x_k)] \leqslant \frac{\psi_0(x_0)-\psi_0^*}{K}$, the result immediately follows. □

Note that even though Assumption 4.2.1 along with (4.33) and (4.34) yields sufficient conditions to guarantee the convergence of the inexact proximal point method, the applicability of Assumption 4.2.1 is limited for the following reasons. First, the optimality criteria of x_k, i.e., relations (4.33) and (4.34) are difficult to verify algorithmically in general since one does not know x_k^*. Second, in order to ensure such conditions, one needs to develop algorithms satisfying convergence of x_k to x_k^*. The ConEx method provided in Chapter 3 exhibits this type of convergence for solving strongly convex function constrained problem (4.9).

However, as in the previous subsection, we can use the condition (4.26) to obtain uniform bounds on $\|y_k^*\|_1$ for Algorithm 3 as well. In particular, the uniform boundedness result of Lemma

4.2.9 is applicable for $\|y_k^*\|_1$ of Algorithm 3 as we never used optimality of x_k in the proof of Lemma 4.2.9. In fact, (4.26) ensures feasibility of the subproblem (4.9) for any $x_{k-1} \in X$. Hence this condition is sufficient for ensuring two core assumptions required for analyzing convergence rates of Algorithm 3: feasibility of (4.9) and boundedness of $\|y_k\|_1$. In this case, we only need to assume that x_k satisfies the optimality gap and constraint violation as given in Definition 3.3.1.

We are now ready to show the convergence result for Algorithm 3.

Theorem 4.2.14 *In Algorithm 3, suppose that Assumption 4.2.3 holds such that $\|y_k^*\|_1 \leqslant B$. Moreover, assume that the definition of x_k in Algorithm 3 is given by*

$$x_k \leftarrow \textit{ a stochastic}(\delta_k, \bar{\delta}_k)\textit{-optimal solution (c.f. Definition 3.3.1) of } (4.9). \tag{4.45}$$

Then $x_{\hat{k}}$ is a stochastic $(\varepsilon_K, \bar{\varepsilon}_K)$-KKT point of Problem (3.1) *with*

$$\varepsilon_K = \max\{2L_\omega, 8L_\omega^2(\mu_0 + \mu_{\max}B)\}\tfrac{\Gamma_K}{K}, \quad \textit{and } \bar{\varepsilon}_K = \tfrac{2}{\mu_0 K}\Omega_K, \tag{4.46}$$

where $\mu_{\max} := \max_{i\in[m]}\mu_i$, $\Gamma_K := \Delta_{\psi_0} + B\bar{\Delta}_0 + \Omega_K$, $\Delta_{\psi_0} := \psi_0(x_0) - \min_{x\in X}\psi_0(x)$, $\bar{\Delta}_0 = \|[\psi(x_0)]_+\|_2$ and $\Omega_K = \sum_{k=1}^K \delta_k + B\sum_{k=1}^K \bar{\delta}_k$.

Proof. Let $\Delta_k = \psi_0(x_k; x_{k-1}) - \psi_0(x_k^*; x_{k-1})$ and $\bar{\Delta}_k = \|\big[\psi(x_k; x_{k-1})\big]_+\|_2$. Using Definition 3.3.1 we have $\mathbb{E}[|\Delta_k|] \leqslant \delta_k$ and $\mathbb{E}[\bar{\Delta}_k] \leqslant \bar{\delta}_k$. In view of Lemma 4.2.3 and the strong convexity of $\psi_0(; x_{k-1})$ and $\psi(; x_{k-1})$, we have

$$\begin{aligned}\psi_0(x; x_{k-1}) + \textstyle\sum_{i=1}^m y_k^{*(i)}\psi_i(x; x_{k-1}) &\geqslant \psi_0(x_k^*; x_{k-1}) + (\mu_0 + \mu^T y_k^*)W(x, x_k^*)\\ &= \psi_0(x_k; x_{k-1}) - \Delta_k + \big(\mu_0 + \mu^T y_k^*\big)\, W(x, x_k^*)\\ &= \psi_0(x_k) + 2\mu_0 W(x_k, x_{k-1}) - \Delta_k + \big(\mu_0 + \mu^T y_k^*\big)\, W(x, x_k^*).\end{aligned} \tag{4.47}$$

Setting $x = x_k$ in (4.47) yields

$$\psi_0(x_k; x_{k-1}) + \textstyle\sum_{i=1}^m y_k^{*(i)}\psi_i(x_k; x_{k-1}) \geqslant \psi_0(x_k^*; x_{k-1}) + (\mu_0 + \mu^T y_k^*)W(x_k, x_k^*)$$

Setting $k = \widehat{k}$ in the above relation and taking expectation, we have

$$\begin{aligned}\mathbb{E}[\|x_{\widehat{k}} - x^*_{\widehat{k}}\|^2] \leqslant 2\mathbb{E}[W(x_{\widehat{k}}, x^*_{\widehat{k}})] &\leqslant \tfrac{2}{\mu_0 K}\textstyle\sum_{k=1}^K \mathbb{E}[\psi_0(x_k; x_{k-1}) - \psi_0(x^*_k;) + \sum_{i=1}^m y_k^{*(i)}\psi_i(x_k; x_{k-1})] \\ &\leqslant \tfrac{2}{\mu_0 K}\textstyle\sum_{k=1}^K \mathbb{E}[\psi_0(x_k; x_{k-1}) - \psi_0(x^*_k; x_{k-1}) + \sum_{i=1}^m y_k^{*(i)}[\psi_i(x_k; x_{k-1})]_+] \\ &\leqslant \tfrac{2}{\mu_0 K}\textstyle\sum_{k=1}^K \mathbb{E}[\Delta_k + B\bar{\Delta}_k] \\ &\leqslant \tfrac{2}{\mu_0 K}\textstyle\sum_{k=1}^K (\delta_k + B\bar{\delta}_k).\end{aligned}$$

where the third inequality above is due to the Cauchy-Schwarz inequality and the boundedness of $\|y^*_k\|_2$: $\|y^*_k\|_2 \leqslant \|y^*_k\|_1 \leqslant B$.

Analogously, by setting $x = x_{k-1}$ in (4.47) and noticing $\psi_0(x_{k-1}; x_{k-1}) = \psi_0(x_{k-1})$ we have

$$\begin{aligned}\psi_0(x_{k-1}) + B\bar{\Delta}_{k-1} &\geqslant \psi_0(x_{k-1}; x_{k-1}) + \|y^*_k\|_2\bar{\Delta}_{k-1} \\ &\geqslant \psi_0(x_{k-1}; x_{k-1}) + \textstyle\sum_{i=1}^m y_k^{*(i)}\psi_i(x_{k-1}; x_{k-1}) \\ &\geqslant \psi_0(x^*_k; x_{k-1}) + \left(\mu_0 + \mu^T y^*_k\right) W(x_{k-1}, x^*_k) \\ &\geqslant \psi_0(x_k) - \Delta_k + 2\mu_0 W(x_k, x_{k-1}) + \left(\mu_0 + \mu^T y^*_k\right) W(x_{k-1}, x^*_k).\end{aligned} \tag{4.48}$$

Here the second inequality use the following property: for $k > 1$,

$$\textstyle\sum_{i=1}^m y_k^{*(i)}\psi_i(x_{k-1}; x_{k-1}) \leqslant \sum_{i=1}^m [y_k^{*(i)}\psi_i(x_{k-1}; x_{k-1})]_+ \leqslant \sum_{i=1}^m y_k^{*(i)}[\psi_i(x_{k-1}; x_{k-1})]_+ \leqslant \|y^*_k\|_2\bar{\Delta}_{k-1}, \tag{4.49}$$

and $\sum_{i=1}^m y_1^{*(i)}\psi_i(x_0; x_0) = \sum_{i=1}^m y_1^{*(i)}\psi_i(x_0) \leqslant \|y^*_1\|_2\bar{\Delta}_0$.

Summing up the inequality (4.48) for $k = 1, \ldots, K$, we obtain

$$\begin{aligned}2\mu_0\textstyle\sum_{k=1}^K W(x_k, x_{k-1}) + \sum_{k=1}^K \left(\mu_0 + \mu^T y^*_k\right) W(x_{k-1}, x^*_k) & \\ \leqslant \psi_0(x_0) - \psi_0(x_K) + \textstyle\sum_{k=1}^K \Delta_k + B\sum_{k=1}^K \bar{\Delta}_{k-1} & \\ \leqslant \Delta_f + \textstyle\sum_{k=1}^K \Delta_k + B\sum_{k=1}^K \bar{\Delta}_{k-1}, &\end{aligned} \tag{4.50}$$

Furthermore, due to the KKT condition for (4.9), we have

$$d\big(\partial\psi_0(x_k^*; x_{k-1}) + \textstyle\sum_{i=1}^m y_k^{*(i)}\partial\psi_i(x_k^*; x_{k-1}) + N_X(x_k^*), \mathbf{0}\big) = 0.$$

Plugging the definition of $\partial\psi_0(x; x_{k-1})$ and $\partial\psi_i(x; x_{k-1}), i \in [m]$, into the above inequality yields

$$d\big(\partial\psi_0(x_k^*) + \textstyle\sum_{i=1}^m y_k^{*(i)}\partial\psi_i(x_k^*) + 2(\mu_0 + \mu^T y_k^*)(\nabla\omega(x_k^*) - \nabla\omega(x_{k-1})) + N_X(x_k^*), \mathbf{0}\big) = 0. \quad (4.51)$$

Let $\widehat{k}$ be the random index from $1, \ldots, K$. Then, in view of (4.51), (4.50) and bound on $\|y_k^*\|_1$, we have

$$\begin{aligned}
&\mathbb{E}\big[d\big(\partial\psi_0(x_{\widehat{k}}^*) + \textstyle\sum_{i=1}^m y_{\widehat{k}}^{*(i)}\partial\psi_i(x_{\widehat{k}}^*) + N_X(x_{\widehat{k}}^*), \mathbf{0}\big)^2\big] \\
&\quad = \tfrac{1}{K}\mathbb{E}\Big\{\textstyle\sum_{k=1}^K d(\partial\psi_0(x_k^*) + \sum_{i=1}^m y_k^{*(i)}\partial\psi_i(x_k^*) + N_X(x_k^*), \mathbf{0})^2\Big\} \\
&\quad \leqslant \tfrac{4}{K}\mathbb{E}\Big\{\textstyle\sum_{k=1}^K (\mu_0 + \mu^T y_k^*)^2\|\nabla\omega(x_k^*) - \nabla\omega(x_{k-1})\|^2\Big\} \\
&\quad \leqslant \tfrac{8L_\omega^2(\mu_0+\mu_{\max}B)}{K}\mathbb{E}\Big\{\textstyle\sum_{k=1}^K (\mu_0 + \mu^T y_k^*)W(x_{k-1}, x_k^*)\Big\} \\
&\quad \leqslant \tfrac{8L_\omega^2(\mu_0+\mu_{\max}B)}{K}\Big[\Delta_f + \bar{\Delta}_0 + \textstyle\sum_{k=1}^K \delta_k + B\sum_{k=2}^K \bar{\delta}_{k-1}\Big] \\
&\quad \leqslant \tfrac{8L_\omega^2(\mu_0+\mu_{\max}B)}{K}\Gamma_K
\end{aligned} \quad (4.52)$$

Moreover, using the complimentary slackness for the subproblem and the relation (4.50), we have

$$\begin{aligned}
\textstyle\sum_{k=1}^K\sum_{i=1}^m |y_k^{*(i)}\psi_i(x_k^*)| &= 2\textstyle\sum_{k=1}^K (\mu^T y_k^*)W(x_k^*, x_{k-1}) \\
&\leqslant 2L_\omega\textstyle\sum_{k=1}^K (\mu^T y_k^*)W(x_{k-1}, x_k^*) \\
&\leqslant 2L_\omega\big[\Delta_f + \textstyle\sum_{k=1}^K \Delta_k + B\sum_{k=1}^K \bar{\Delta}_{k-1}\big].
\end{aligned} \quad (4.53)$$

Therefore

$$\mathbb{E}\Big[\textstyle\sum_{i=1}^m |y_{\widehat{k}}^{*(i)}\psi_i(x_{\widehat{k}}^*)|\Big] = \tfrac{1}{K}\mathbb{E}\Big[\textstyle\sum_{k=1}^K\sum_{i=1}^m |y_k^{*(i)}\psi_i(x_k^*)|\Big] \leqslant \tfrac{2L_\omega}{K}\Gamma_K.$$

Hence we conclude the proof. □

Remark 4.2.15 *We should note that when $\psi_i, i \in [m]$, are convex functions then we can obtain a variant of Algorithm 3 where x_k is a (stochastic) $(\delta_k, \bar{\delta}_k)$-optimal solution of* (4.32)*. For this variant of Algorithm 3, we can easily obtain the result of Theorem 4.2.14 under Assumption 4.2.3. Moreover, since constraints remain same in* (4.32) *for all $k \geqslant 1$, we just need Slater condition to ensure uniform boundedness of $\|y_k\|_1$.*

In the following corollary, we state an immediate consequence of Theorem 4.2.14 as well as the final complexity when using the ConEx method as subroutine to solve subproblem 4.9. Before proceeding to the details of the corollary, we need to properly redefine B such that it satisfies $B \geqslant \max\{\|y_k^*\|_1, \|y_k^*\|_2 + 1\}$. This allows the use of B in the sense of Theorem 4.2.14 as well as in the stepsize policy for the ConEx method in (3.15).

Corollary 4.2.16 *Under the assumptions of Theorem 4.2.14, suppose that in Algorithm 3, we set $\delta_k = c\bar{\delta}_k$ for some $c > 0$, and $\bar{\delta}_k = \varepsilon/(2c_1c_2)$, where*

$$\begin{aligned} c_1 &= \max\big\{2L_w, 8L_w^2(\mu_0 + \mu_{\max}B)\big\} \\ c_2 &= c + B \end{aligned} \tag{4.54}$$

Then after running at most $K = 2c_1(\Delta_f + B\bar{\Delta}_0)/\varepsilon$ iterations, we obtain an $(\varepsilon, \frac{2\varepsilon}{\mu_0 c_1})$-KKT point of Problem (3.1)*. In particular, if we run Algorithm 1 for subproblem* (4.9)*, then we obtain an $(\varepsilon, \frac{2\varepsilon}{\mu_0 c_1})$-KKT point in $O(\frac{1}{\varepsilon}T_\varepsilon)$ iterations, where T_ε is defined in* (3.19)*.*

Proof. Suppose δ_k and $\bar{\delta}_k$ are constants throughout Algorithm 3. Then, according to (4.46), we have $\varepsilon_K \leqslant c_1\Gamma_K/K$. Choosing given values of $\delta_k, \bar{\delta}_k$ and K, we have

$$\varepsilon_K \leqslant c_1\frac{\Gamma_K}{K} = c_1\left[\frac{\Delta_f + B\bar{\Delta}_0}{K} + (c+B)\bar{\delta}\right] = c_1\left[\frac{\varepsilon}{2c_1} + c_2\frac{\varepsilon}{2c_1c_2}\right] = \varepsilon.$$

Moreover, we have

$$\bar{\varepsilon}_K = \tfrac{2}{\mu_0 K}\Omega_K \leqslant \tfrac{2}{\mu_0 K}\Gamma_K \leqslant \tfrac{2\varepsilon}{\mu_0 c_1}.$$

Now noting that $\delta_k = \bar{\delta}_k = O(\varepsilon)$ is a constant and using Corollary 3.3.2, we obtain $(\delta_k, \bar{\delta}_k)$-approximate solution of subproblem (4.9) in T_ε iterations. Noting the definition K in the statement of the corollary, we conclude the proof. □

In the above corollary, we assume that the subproblem (4.9) is solved by using the ConEx method. In particular, if $\chi_i(x)$ is a simple function such that we can compute **prox** operator in (3.8) for functions $\mu_i W(x, x_{k-1}) + \chi_i(x),\ i = 1, \ldots, m$, efficiently, then we solve each subproblem in the smooth strongly convex setting, since $f_i, i = 1, \ldots, m$ are smooth functions. Otherwise, we must include the nonsmooth convex function $\chi_i(x)$ in totality (or part thereof) with f_i, and then we can assume $\mu_i W(x, x_{k-1})$ is a simple function. In this case, we solve the subproblems in a nonsmooth strongly convex setting. We can derive from Corollary 4.2.16 and the definition of T_ε in (3.19) the final complexity bounds for different problem settings.

- **Smooth nonconvex case**: In this case, T_ε can be bounded $O(1/\varepsilon^{1/2})$ in the deterministic case, $O(1/\varepsilon)$ in the semi-stochastic case and $O(1/\varepsilon^2)$ in the fully-stochastic case. Hence, in view of Corollary 4.2.16, we can compute an $(\varepsilon, 2\varepsilon/(\mu_0 c_1))$-KKT point of the nonconvex problem (3.1) in $O(1/\varepsilon^{3/2})$, $O(1/\varepsilon^2)$, and $O(1/\varepsilon^3)$ iterations for the deterministic case, semi-stochastic case and fully-stochastic cases, respectively.

- **Nonsmooth nonconvex case**: In this case, T_ε can be bounded by $O(1/\varepsilon)$ in the deterministic case, $O(1/\varepsilon)$ in the semi-stochastic case and $O(1/\varepsilon^2)$ in the fully-stochastic case. Hence, in view of Corollary 4.2.16, we can compute an $(\varepsilon, 2\varepsilon/(\mu_0 c_1))$-KKT point of the nonconvex problem (3.1) in $O(1/\varepsilon^2)$ iterations for the deterministic and semi-stochastic cases, and $O(1/\varepsilon^3)$ iterations for the fully-stochastic case.

Note that the dependence of these complexity bounds on different problem parameters can be made more precise in view of the definition of T_ε in (3.19).

4.3 Proofs of Auxiliary Results

4.3.1 Proof of Proposition 4.2.1

Let us denote

$$\bar{\psi}_0(x) := \psi_0(x) + \tfrac{\mu_0}{2}\|x - x^*\|_2^2,$$
$$\bar{\psi}_i(x) := \psi_i(x) + \tfrac{\mu_i}{2}\|x - x^*\|_2^2.$$

It is easy to see that $\bar{\psi}_0(x)$ and $\bar{\psi}_i(x),\ i \in [m]$, are convex functions. Moreover, their respective subdifferentials can be written as

$$\partial\bar{\psi}_0(x) = \{\nabla f_0(x) + \mu_0(x - x^*)\} + \partial\chi_0(x),$$
$$\partial\bar{\psi}_i(x) = \{\nabla f_i(x) + \mu_i(x - x^*)\} + \partial\chi_i(x).$$

Consider the constrained convex optimization problem:

$$\begin{aligned} \min_{x\in X} \quad & \bar{\psi}_0(x) \\ \text{s.t.} \quad & \bar{\psi}_i(x) \leqslant 0, \qquad i \in [m]. \end{aligned} \tag{4.55}$$

Note that x^* is a feasible solution of this problem. For sake of this proof, define $\Psi_k(x) := \bar{\psi}_0(x) + \frac{k}{2}\sum_{i=1}^m \big[\bar{\psi}_i(x)\big]_+^2 + \frac{1}{2}\|x - x^*\|_2^2$. Let $S = \{x : \|x - x^*\|_2 < \varepsilon\}$ for some $\varepsilon > 0$ such that any $x \in S$ which is feasible for (4.55) satisfies $\bar{\psi}_0(x) \geqslant \bar{\psi}_0(x^*)$. Let $x_k := \operatorname{argmin}_{x\in S\cap X} \Psi_k(x)$. Note that as $k \to \infty$ then due to the optimality of x_k and existence of $x^* \in S \cap X$, we have $\lim_{k\to\infty} \bar{\psi}(x_k) \leqslant 0$. Since $\lim_{k\to\infty} x_k$ is feasible for (4.55) so we conclude that $x_k \to x^*$. Hence there exists $\bar{k}$ such that for all $k > \bar{k}$, $x_k \in \operatorname{int}(S)$. So for such k we can write the following first-order criterion for convex optimization ($[\bar{\psi}_i]_+^2$ is a convex function):

$$\mathbf{0} \in N_X(x_k) + \partial\bar{\psi}_0(x_k) + k[\bar{\psi}(x_k)]_+\partial\bar{\psi}(x_k) + x_k - x^*.$$

This implies that x_k is also the optimal solution of

$$\min_{x\in X} \bar{\psi}_0(x) + k\,[\bar{\psi}(x_k)]_+^T\psi(x) + \tfrac{1}{2}\|x - x^*\|^2.$$

For simplicity, let us denote $v_k = k\,[\bar{\psi}(x_k)]_+^T$. Due to the optimality of x_k of solving the above, we have

$$\bar{\psi}_0(x_k) + v_k^T\bar{\psi}(x_k) + \tfrac{1}{2}\|x_k - x^*\|^2 \leqslant \bar{\psi}_0(x) + v_k^T\bar{\psi}(x) + \tfrac{1}{2}\|x - x^*\|^2, \quad \forall x \in X. \tag{4.56}$$

We claim that $\{v_k\}$ is a bounded sequence. Indeed, if this is true, then we can find a convergent subsequence $\{i_k\}$ with $\lim_{k\to\infty} v_{i_k} = v^*$. Taking $k \to \infty$ in (4.56), we have

$$\limsup_{k\to\infty} \bar{\psi}_0(x_{i_k}) + v^{*T}\bar{\psi}(x^*) \leqslant \bar{\psi}_0(x) + v^{*T}\bar{\psi}(x) + \tfrac{1}{2}\|x - x^*\|^2, \qquad \forall x \in X. \tag{4.57}$$

Placing $x = x^*$, we have $\bar{\psi}_0(x^*) \geqslant \limsup \bar{\psi}_0(x_{i_k})$, thus $\lim_{k\to\infty} \bar{\psi}_0(x_{i_k}) = \bar{\psi}_0(x^*)$ based on the lower semicontinuity of $\bar{\psi}_0$. In view of this discussion, x^* optimizes the right side of (4.57). Thus, applying the first order criterion, we have

$$0 \in \partial\bar{\psi}_0(x^*) + \sum_{i\in[m]} v^{(i)^*}\partial\bar{\psi}(x^*) + N_X(x^*).$$

It remains to apply $\partial\bar{\psi}_0(x^*) = \partial\psi_0(x^*)$ and $\partial\bar{\psi}_i(x^*) = \partial\psi_i(x^*)$.

In addition, to prove complimentary slackness, it suffices to show when $\bar{\psi}_i(x^*) = \psi_i(x^*) < 0$, we must have $v^{(i)^*} = 0$. Since x_k converges to x^* and $\bar{\psi}_i$ is continuous, there exists some $\exists k_0 > 0$, such that $\bar{\psi}_i(x_{i_k}) < 0$ when $k > k_0$. Hence $v_{i_k}^{(i)^*} = 0$ by its definition. Taking the limit, we have $v^{(i)^*} = 0$.

It remains to show the missing piece, that $\{v_k\}$ is a bounded sequence. We will prove by contradiction. If this is not true, we may assume $\lim_{k\to\infty} \|v_k\| = \infty$, passing to a subsequence if necessary. Moreover, define $y_k = v_k/\|v_k\|$, since y_k is a unit vector, it has some limit point, let us

assume $\lim_{k\to\infty} y_{j_k} = y^*$ for a subsequence $\{j_k\}$. Dividing both sides of (4.56) by $\|v_k\|$ and then passing it to the subsequence $\{j_k\}$, we have

$$\bar{\psi}_0(x_{j_k})/\|v_{j_k}\| + y_{j_k}^T\bar{\psi}(x_{j_k}) + \tfrac{1}{2\|v_{j_k}\|}\|x_{j_k} - x^*\|^2 \leqslant \bar{\psi}_0(x) + y_{j_k}^T\bar{\psi}(x) + \tfrac{1}{2\|v_{j_k}\|}\|x - x^*\|^2, \qquad \forall x \in X.$$

Taking $k \to \infty$, we have

$$y^{*T}\bar{\psi}(x^*) \leqslant y^{*T}\bar{\psi}(x), \qquad \forall x \in X.$$

Since subsequence x_{j_k} converges to x^* and $\bar{\psi}_i$ is continuous, we see that $\bar{\psi}_i(x_{j_k}) < 0$ for any $i \notin \mathcal{A}(x^*)$ for $k \geqslant k_0$. This implies $y_{j_k} = j_k\big[\bar{\psi}_i(x_{j_k})\big]_+ = 0$ for all $k \geqslant k_0$ and for all $i \notin \mathcal{A}(x^*)$. So we must have $\mathbf{0} \in N_X(x^*) + \sum_{i\in\mathcal{A}(x^*)} y^{*(i)}\partial\psi_i(x^*)$. Let $u \in N_X(x^*)$ and $g_i(x^*) \in \partial\psi_i(x^*), i \in \mathcal{A}(x^*)$ be such that

$$u + \textstyle\sum_{i\in\mathcal{A}(x^*)} y^{*(i)} g_i(x^*) = \mathbf{0}.$$

Then we can derive a contradiction by using Assumption 4.2.1 (MFCQ). Assume that z satisfies MFCQ (4.2.1). Therefore, we have

$$\begin{aligned} 0 = z^T u + \textstyle\sum_{i\in\mathcal{A}(x^*)} y^{*(i)} z^T g_i(x^*) &\leqslant \textstyle\sum_{i\in\mathcal{A}(x^*)} y^{*(i)} z^T g_i(x^*) \\ &\leqslant \textstyle\sum_{i\in\mathcal{A}(x^*)} y^{*(i)} \max_{v\in\partial\psi_i(x^*)} z^T v < 0, \end{aligned}$$

where first inequality follows since $z \in -N_X^*(x^*)$ and $u \in N_X(x^*)$ hence $z^T u \leqslant 0$, second inequality follows due to the fact that $y^{*(i)} \geqslant 0$ and $g_i(x^*) \in \partial\psi_i(x^*)$ and last strict inequality follows since (4.2.1) and $y^{*(i)} > 0$ for at least one $i \in \mathcal{A}(x^*)$.

CHAPTER 5

LEVEL PROXIMAL POINT METHOD FOR NONCONVEX SPARSE CONSTRAINED OPTIMIZATION

In the previous chapter, we saw an inexact proximal point method for solving nonconvex function constrained optimization problem. In order to obtain the convergence to a KKT-point, we required that the sequence of Lagrange multipliers for the convex subproblem generated by proximal point method remain bounded. We resorted to strong feasibility assumption to ensure such a bound. In essence, strong feasibility assumption gives us a guarantee that all the convex subproblem generated at any point in the set X has a strictly feasible point which can be used to bound the Lagrange multiplier. In this chapter, we consider a class of problems which lie in the larger family of nonsmooth nonconvex function constrained optimization problems in Chapter 4 and do not require this strong feasibility assumption for ensuring a bound on the Lagrange multiplier. In particular, we will consider constrained optimization problems with nonconvex (and nonsmooth) sparsity inducing constraints. We will show convergence to a KKT-point for objectives which can be convex or nonconvex, smooth or nonsmooth and deterministic or stochastic under MFCQ constraint qualification without requiring strong feasibility. Our assumptions on the structure of the constraint is fairly general and are satisfied by variety of sparsity inducing constraints in the literature. Moreover, our convergence rates will be faster compared to those obtained in Chapter 4 due to an effective projection mechanism.

5.1 Nonconvex Sparse Constrained Optimization

Recent years have witnessed a great deal of work on the sparse optimization arising from machine learning, statistics and signal processing. A fundamental challenge in this area lies in finding the

best set of size k out of a total of d ($k < d$) features to form a parsimonious fit to the data:

$$\min\ \psi(x), \quad \text{subject to} \quad \|x\|_0 \leqslant k, x \in \mathbb{R}^d. \tag{5.1}$$

However, due to the discontinuity of $\|\cdot\|_0$ norm[1], the above problem is intractable when there is no other assumptions.

5.1.1 Existing models

To bypass the difficulty of handling ℓ_0-norm, a popular approach is to replace the ℓ_0-norm by the ℓ_1-norm, giving rise to an ℓ_1-constrained or ℓ_1-regularized problem. A notable example is the Lasso ([105]) approach for linear regression and its regularized variant

$$\min \|b - Ax\|_2^2, \qquad \text{subject to } \|x\|_1 \leqslant \tau, x \in \mathbb{R}^d; \tag{5.2}$$

$$\min \|b - Ax\|_2^2 + \lambda \|x\|_1. \tag{5.3}$$

Due to the Lagrange duality theory, problem (5.2) and (5.3) are equivalent in the sense that there is a one-to-one mapping between the parameters τ and λ. A substantial amount of literature already exists for understanding the statistical properties of ℓ_1 models ([122, 106, 19, 120, 122]) as well as for the development efficient algorithms when such models are employed ([33, 9, 83, 111]).

In spite of their success, ℓ_1 models can be suboptimal due to the looseness of the convex relaxation. To overcome this issue, a large body of the recent work proposes to replace the ℓ_1-penalty in (5.3) by a nonconvex function $g(x)$ to obtain sharper approximation of the ℓ_0-norm:

$$\min\ \psi(x) + \lambda g(x). \tag{5.4}$$

Despite the favorable statistical properties ([35, 119, 20, 121]), nonconvex models have posed a great challenge for optimization algorithms and has been increasingly an important issue ([43, 42,

[1]Note that $\|\cdot\|_0$ is not a norm in mathematical sense. Indeed, $\|x\|_0 = \|tx\|_0$ for any nonzero t.

49, 103]).

5.1.2 A new model for nonconvex sparse constrained optimization

Most of these works studied the regularized version. However, it is often favorable to consider the following constrained form:

$$\min \ \psi(x), \quad \text{subject to} \quad g(x) \leqslant \eta, x \in \mathbb{R}^d \tag{5.5}$$

because the sparsity of solutions is imperative in many applications of statistical learning and the constrained form in (5.5) explicitly imposes such a requirement. Therefore, it is natural to ask *whether we can provide an efficient algorithm for problem* (5.5). The continuous nonconvex relaxation (5.5) of the ℓ_0-norm in (5.1), albeit a straightforward one, was not studied in the literature. We suspect that to be the case due to the difficulty in handling nonconvex constraints algorithmically. There are two theoretical challenges: First, since the regularized form (5.4) and the constrained form (5.5) are not equivalent due to the nonconvexity of $g(x)$, we cannot bypass (5.5) by solving problem (5.4) instead. Second, the nonconvex function $g(x)$ can be nonsmooth especially for the sparsity applications, presenting a substantial challenge for classic nonlinear programming methods, e.g., augmented Lagrangian methods and penalty methods (see [12]) which assumes that functions are continuously differentiable.

5.1.3 New algorithm for the proposed new model

In this chapter, we study a newly proposed nonconvex constrained model (5.5) from an algorithmic point of view. In particular, we present a novel level-constrained proximal point (LCPP) method for problem (5.5) where the objective ψ can be either deterministic/stochastic, smooth/nonsmooth and convex/nonconvex and the constraint g models a variety of sparsity inducing nonconvex constraints proposed in the literature. The key idea is to translate problem (5.5) into a sequence of convex subproblems where $\psi(x)$ is *convexified* using a proximal point quadratic term and $g(x)$ is *majorized*

by a convex function $\widetilde{g}(x)[\geqslant g(x)]$. Note that $\{\widetilde{g}(x) \leqslant \eta\}$ is a convex subset of the nonconvex set $\{g(x) \leqslant \eta\}$.

We show that starting from a strict feasible point[2], LCPP traces a feasible solution path with respect to the set $\{g(x) \leqslant \eta\}$. We also show that LCPP generates convex subproblems for which bounds on the optimal Lagrange multiplier (or the optimal dual) can be provided under a mild and a well-known constraint qualification. This bound on the dual and the proximal point update in the objective allows us to prove asymptotic convergence to the KKT points of the problem (5.5).

While deriving the complexity, we consider the inexact LCPP method that solves convex subproblems approximately. We show that the constraint, $\widetilde{g}(x) \leqslant \eta$, has an efficient projection algorithm. Hence, each convex subproblem can be solved by projection-based first-order methods. This allows us to be feasible even when the solution reaches arbitrarily close to the boundary of the set $\{g(x) \leqslant \eta\}$ which entails that the bound on the dual mentioned earlier works in the inexact case too. Moreover, efficient projection-based first-order method for solving the subproblem helps us get an accelerated convergence complexity of $O(1/\varepsilon)[O(1/\varepsilon^2)]$ gradient [stochastic gradient] in order to obtain an ε-KKT point. In particular, refer to Table 5.1. We see that in the case where objective is smooth and deterministic, we obtain convergence rate of $O(1/\varepsilon)$ whereas for nonsmooth and/or stochastic objective we obtain convergence rate of $O(1/\varepsilon^2)$. This complexity is nearly the same as that of the gradient [stochastic gradient] descent for the regularized problem (5.4) of the respective type.

Remarkably, this convergence rate is better than black-box nonconvex function constrained optimization methods proposed in the literature recently ([16, 64]). We will discuss this in more detail soon. For now, note that the convergence of gradient descent does not ensure a bound on the infeasibility of the constraint g, whereas the KKT criterion requires feasibility on top of stationarity. Moreover, such a bound cannot be ensured theoretically due to the absence of duality. Hence, our algorithm provides additional guarantees without paying much in the complexity.

We perform numerical experiments to measure the efficiency of our LCPP method and the

[2]Origin is always strictly feasible for sparsity inducing constraints and can be chosen as a starting point.

Table 5.1: Convergence rates of LCPP for problem (5.5) when the objective can be either convex or nonconvex, smooth or nonsmooth and deterministic or stochastic

	Convex (5.5)		Nonconvex (5.5)	
Cases	Smooth	Nonsmooth	Smooth	Nonsmooth
Deterministic	$O(1/\varepsilon)$	$O(1/\varepsilon^2)$	$O(1/\varepsilon)$	$O(1/\varepsilon^2)$
Stochastic	$O(1/\varepsilon^2)$	$O(1/\varepsilon^2)$	$O(1/\varepsilon^2)$	$O(1/\varepsilon^2)$

effectiveness of the new constrained model (5.5). First, we show that our algorithm has running time performance which is competitive against open-source solvers, e.g., DCCP [98]. Second, we also compare the effectiveness of our constrained model with respect to the existing convex and nonconvex regularization models in the literature. Our numerical experiments show promising results compared to ℓ_1-regularization model 5.3 and has competitive performance with respect to recently developed algorithm for nonconvex regularization model 5.4 (see [42]). Given that this is the first study in the development of algorithms for the constrained model, we believe empirical study of even more efficient algorithms solving problem (5.5) may be of independent interest and can be pursued in the future.

5.1.4 Existing methods similar to the proposed algorithm

There is a growing interest in using convex majorization for solving nonconvex optimization with nonconvex function constraints.

Typical frameworks include difference-of-convex (DC) programming ([104]), majorization-minimization ([102]) to name a few. Considering the substantial literature, we emphasize the most relevant work to our current paper. Scutari et al. [95] proposed general approaches to majorize nonconvex constrained problems and include (5.5) as a special case. They require exact solutions of the subproblems and prove asymptotic convergence which is prohibitive for large-scale optimization. Shen et al. [98] proposed a disciplined convex-concave programming (DCCP) framework for a class of DC programs in which (5.5) is a special case. Their work is empirical and does not provide specific convergence results.

The more recent works [16, 64] considered a type of proximal point method in which they

add a large enough quadratic proximal term into both objective and constraint in order to obtain a convex subproblem. This convex function constrained subproblem can be solved by oracles whose output solution might have small infeasibility. Moreover these oracles have weaker convergence rates. Complexity results proposed in these works, when applied to problem (5.5), entail $O(1/\varepsilon^{3/2})$ iterations for obtaining an ε-KKT point under a *strong feasibility* constraint qualification. In similar setting, we show faster convergence result of $O(1/\varepsilon)$. This due to the fact that our oracle for solving the subproblem is more efficient than those used in their paper. We can obtain such an oracle due to the availability of efficient projection onto convex surrogate constraint. Moreover, our convergence results hold under a well-known constraint qualification which is weaker compared to *strong feasibility* since our oracle outputs a feasible solution whereas they can get a solution which is slightly infeasible.

5.2 Level Constrained Proximal Point Method

Given this background, now we focus our attention to the main problem at hand. Our main goal is to solve problem (5.5). We make Assumption 5.2.1 throughout the paper.

Assumption 5.2.1 *1. $\psi(x)$ is a continuous and possibly nonsmooth nonconvex function satisfying:*

$$\psi(x) \geqslant \psi(y) + \langle \psi'(y), x - y \rangle - \tfrac{\mu}{2}\|x - y\|_2^2. \tag{5.6}$$

2. $g(x)$ is a nonsmooth nonconvex function of the form $g(x) = \lambda\|x\|_1 - h(x)$, where $h(x)$ is convex and continuously differentiable.

The Lagrangian function for problem (5.5) is defined as $\mathcal{L}(x, y) = \psi(x) + yg(x)$ where $y \geqslant 0$. For nonconvex nonsmooth function $g(x)$ in the form of (5.2), we denote its *subdifferential*[3] by $\partial g(x) = \partial(\lambda\|x\|_1) - \nabla h(x)$. For this definition of subdifferential, we consider the following KKT condition:

[3]Various subdifferentials exist in the literature for nonconvex optimization problem. Here, we use subdifferential Definition 3.1 in Boob et al. [16] for nonconvex nonsmooth function g.

Table 5.2: Examples of constraint function $g(x) = \lambda\|x\|_1 - h(x)$.

Function $g(x)$	Parameter λ	Function $h(x)$
MCP[119]	λ	$h_{\lambda,\theta}(x) = \begin{cases} \frac{x^2}{2\theta} & \text{if } \lvert x\rvert \leqslant \theta\lambda, \\ \lambda\lvert x\rvert - \frac{\theta\lambda^2}{2} & \text{if } \lvert x\rvert > \theta\lambda. \end{cases}$
SCAD[35]	λ	$h_{\lambda,\theta}(x) = \begin{cases} 0 & \text{if } \lvert x\rvert \leqslant \lambda, \\ \frac{x^2-2\lambda\lvert x\rvert+\lambda^2}{2(\theta-1)} & \text{if } \lambda < \lvert x\rvert \leqslant \theta\lambda, \\ \lambda\lvert x\rvert - \frac{1}{2}(\theta+1)\lambda^2 & \text{if } \lvert x\rvert > \theta\lambda. \end{cases}$
Exp[17]	λ	$h_\lambda(x) = e^{-\lambda\lvert x\rvert} - 1 + \lambda\lvert x\rvert$.
Log[110]	$\frac{\theta}{\log(1+\theta)}$	$h_\theta(x) = \frac{\theta}{\log(1+\theta)}\lvert x\rvert - \frac{\log(1+\theta\lvert x\rvert)}{\log(1+\theta)}$.
$\ell_p(0 < p < 1)$[38]	$\frac{\varepsilon^{1/\theta-1}}{\theta}$	$h_{\varepsilon,\theta}(x) = \frac{\varepsilon^{1/\theta-1}}{\theta}\lvert x\rvert - (\lvert x\rvert + \varepsilon)^{1/\theta}$.
$\ell_p(p < 0)$[92]	$-p\theta$	$h_\theta(x) = -p\theta\lvert x\rvert - 1 + (1 + \theta\lvert x\rvert)^p$.

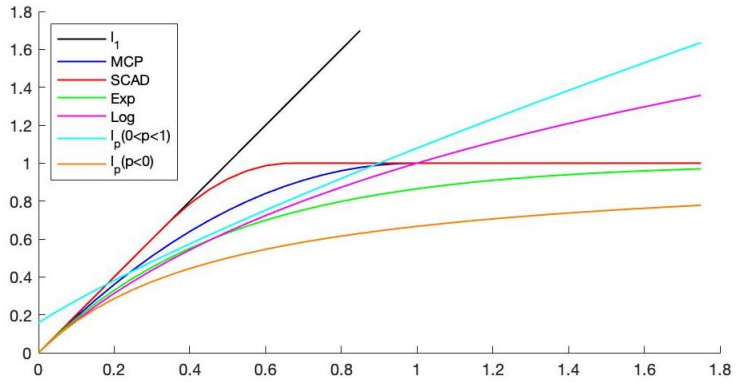


Figure 5.1: Graphs for various constraints along with ℓ_1. For $\ell_p(0 < p < 1)$, we have $\varepsilon = 0.1$.

The KKT condition For Problem (5.5), we say that x is the (stochastic) (ε, δ)- KKT solution if there exists $\bar{x}$ and $\bar{y} \geqslant 0$ such that $g(\bar{x}) \leqslant \eta$, $\mathbb{E}\|x - \bar{x}\|^2 \leqslant \delta$

$$\begin{aligned} \mathbb{E}\left|\bar{y}\left[g(\bar{x}) - \eta\right]\right| &\leqslant \varepsilon \\ \mathbb{E}\left[\operatorname{dist}\left(\partial_x \mathcal{L}(\bar{x}, \bar{y}), 0\right)\right]^2 &\leqslant \varepsilon \end{aligned} \tag{5.7}$$

Moreover, for $\varepsilon = \delta = 0$, we have that $\bar{x}$ is the KKT solution or satisfied KKT condition. If $\delta = O(\varepsilon)$, we refer to this solution as an ε-KKT solution in order to be brief.

It should be mentioned that local or global optimality does not generally imply the KKT condition. However, it is shown to be necessary for optimality when Mangasarian-Fromovitz constraint qualification (MFCQ) holds [16]. Below, we make MFCQ assumption precise:

Assumption 5.2.2 (MFCQ [16]) *Whenever the constraint is active: $g(\bar{x}) = \eta$, there exists a direction z such that* $\max_{v \in \partial g(\bar{x})} v^T z < 0$.

For differentiable g, MFCQ requires existence of z such that $z^T \nabla g(\bar{x}) < 0$, reducing to the classical form of MFCQ [12]. Below, we summarize necessary optimality condition under MFCQ from Chapter 4.

Proposition 5.2.1 (Necessary condition) *Let $\bar{x}$ be a local optimal solution of problem* (5.5)*. If $\bar{x}$ satisfies Assumption 5.2.2, then there exists $\bar{y} \geqslant 0$ such that* (5.7) *holds with $\varepsilon = \delta = 0$.*

Consider the following LCPP method: LCPP method solves sequence of convex subproblems

Algorithm 4 Level constrained proximal point **(LCPP)** method

1: **Input:** $x^0 = \hat{x}$, $\gamma > 0$, $\eta_0 < \eta$
2: **for** $k = 1$ **to** K **do**
3: Set $\eta_k = \eta_{k-1} + \delta_k$;
4: $g_k(x) := \lambda \|x\|_1 - h(x^{k-1}) - \nabla h(x^{k-1})^T (x - x^{k-1})$;
5: Return feasible solution x^k of the problem

$$\min \psi_k(x) = \psi(x) + \tfrac{\gamma}{2} \|x - x^{k-1}\|_2^2, \qquad \text{subject to} \quad g_k(x) \leqslant \eta_k \tag{5.8}$$

6: **end for**

(5.8). In particular, note that $g_k(x)$ majorizes $g(x)$: $g_k(x) \geqslant g(x),\ g_k(x^{k-1}) = g(x^{k-1})$. implying that $\{g_k(x) \leqslant \eta_k\}$ is a convex subset of the original problem. It can also be observed that adding a proximal term in the objective yields ψ_k strongly convex for large enough $\gamma > 0$. In the current form, Algorithm 4 requires a feasible solution of (5.8) and requirement of sequence $\{\eta_k\}$ is left unspecified.

We first make the following assumptions.

Assumption 5.2.3 (Strict feasibility) *There exist sequence $\{\eta_k\}_{k \geqslant 0}$ satisfying:*

1. $\eta_0 < \eta$ and a point $\hat{x}$ of such that $g(\hat{x}) < \eta_0$.

2. The sequence $\{\eta_k\}$ is monotonically increasing and converges to η: $\lim_{k \to \infty} \eta_k = \eta$.

In light of Assumption 5.2.3, starting from a strictly feasible point x^0, Algorithm 4 solves subproblems (5.8) with gradually relaxed constraint levels. This allows us to assert that each subproblem

is strictly feasible. Indeed, we have $g_k(x^k) \leqslant \eta_k \Rightarrow g_{k+1}(x^k) = g(x^k) \leqslant g_k(x^k) \leqslant \eta_k < \eta_{k+1}$. This implies the existence of KKT solution for each subproblem. A formal statement can be found in the appendix. Moreover, all the proofs of our technical results can also be found in the appendix and we just make statements in the main article henceforth.

5.3 Convergence Analysis

First we look at the asymptotic convergence results.

5.3.1 Asymptotic convergence of LCPP method and boundedness of the optimal dual

Our next goal is to establish asymptotic convergence of Algorithm 4 to the KKT points. To this end, we require a uniform boundedness assumption on the Lagrange multipliers. First, we prove asymptotic convergence under this assumption then we justify it under MFCQ. Before precisely stating the convergence results, we make the following boundedness assumption.

Assumption 5.3.1 (Boundedness of dual variables) *There exists $B > 0$ such that $\sup_k \bar{y}^k < B$.*

The following asymptotic convergence theorem is in order.

Theorem 5.3.1 (Convergence to KKT) *Let π_k denotes the randomness of $x^1, x^2, ..., x^{k-1}$. Assume that there exists a $\rho \in [0, \gamma - \mu]$ and a summable nonnegative sequence ζ_k such that*

$$\mathbb{E}[\psi_k(x^k) - \psi_k(\bar{x}^k)|\pi_k] \leqslant \tfrac{\rho}{2}\|\bar{x}^k - x^{k-1}\|_2^2 + \zeta_k. \tag{5.9}$$

Then, under Assumption 5.2.3 and 5.3.1 for any limit point $\widetilde{x}$ of the proposed algorithm, there exists a dual variable $\widetilde{y}$ such that $(\widetilde{x}, \widetilde{y})$ satisfies KKT condition, almost surely.

This theorem shows that any limit point of Algorithm 4 converges to a KKT point. However, it makes the assumption that dual is bounded. Since the optimal dual depends on the convex subproblems (5.8) which are generated dynamically in the algorithm, it is important to justify

Assumption 5.3.1. To this end, we show that Assumption 5.3.1 is satisfied under a well-known constraint qualification.

Theorem 5.3.2 (Boundedness condition) *Suppose Assumption* (5.2.3) *and relation* (5.9) *are satisfied and all limit points of Algorithm 4 exists a.s., and satisfy the MFCQ condition. Then,* $\overline{y}^k$ *is bounded a.s.*

This theorem shows the existence of dual under the MFCQ assumption for all limit points of Algorithm 4. MFCQ is a mild constraint qualification frequently used in the existing literature [12]. In certain cases, we also provide explicit bounds on the dual variables. These bounds quantify how "closely" the MFCQ assumption is violated and provides its effect on the magnitude of the optimal dual. Additional results and discussion in this regard are deferred to the last section. For our purpose now, we assume that the dual variables remain bounded henceforth.

In the next subsection, we show convergence complexity results for the LCPP method.

5.3.2 Complexity of LCPP method

Our goal here is to analyze the complexity of the proposed algorithm. Apart from the negative lower curvature guarantee (5.6) of the objective function, we impose that h has Lipschitz continuous gradients, $\|\nabla h(x) - \nabla h(y)\|_2 \leqslant L_h\|x - y\|_2$. This is satisfied by all functions in Table 5.2. Below, we discuss a general convergence result of LCPP method for original nonconvex problem (5.5).

Theorem 5.3.3 *Suppose Assumption 5.2.3 and 5.3.1 hold such that* $\delta_k = \frac{\eta - \eta_0}{k(k+1)}$ *for all* $k \geqslant 1$. *Let* x^k *satisfy* (5.9) *where* $\rho \in [0, \gamma - \mu]$ *and* $\{\zeta_k\}$ *is a summable nonnegative sequence. Moreover,* x^k *is a feasible solution of the* k*-th subproblem, i.e.,*

$$g_k(x^k) \leqslant \eta_k. \tag{5.10}$$

If $\hat{k}$ is chosen uniformly at random from $\left\lfloor \frac{K+1}{2} \right\rfloor$ to K then there exists a pair $(\bar{x}^{\hat{k}}, \bar{y}^{\hat{k}})$ satisfying

$$\begin{aligned}
\mathbb{E}[\mathrm{dist}(\partial_x \mathcal{L}(\bar{x}^{\hat{k}}, \bar{y}^{\hat{k}}), 0)^2] &\leqslant \tfrac{16(\gamma^2+B^2L_h^2)}{K(\gamma-\mu-\rho)}\left(\tfrac{\gamma-\mu+\rho}{2(\gamma-\mu)}\Delta^0 + Z\right), \\
\mathbb{E}[\bar{y}^{\hat{k}}|g(\bar{x}^{\hat{k}}) - \eta|] &\leqslant \tfrac{2BL_h}{K(\gamma-\mu-\rho)}\left(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2Z\right) + \tfrac{2B(\eta-\eta_0)}{K}, \\
\mathbb{E}[\|x^{\hat{k}} - \bar{x}^{\hat{k}}\|^2] &\leqslant \tfrac{4\rho(\gamma-\mu+\rho)}{K(\gamma-\mu)^2(\gamma-\mu-\rho)}\Delta^0 + \tfrac{8Z}{K(\gamma-\mu-\rho)},
\end{aligned}$$

where, $\Delta^0 := \psi(x^0) - \psi(x^*)$, $Z := \sum_{k=1}^{K}\zeta_k$ and expectation is taken over the randomness of $\widehat{k}$ and solutions x^k, $k = 1, \ldots, K$.

Note that Theorem 5.3.3 assumes that subproblem (5.8) can be solved according to the framework of (5.9) and (5.10). When the subproblem solver is deterministic then we ignore the expectation in (5.9). It is easy to see from the above theorem that for $x^{\hat{k}}$ to be an ε-KKT point, we must have $K = O(1/\varepsilon)$ and ζ_k must be small enough such that Z is bounded above by a constant. The complexity analysis of different cases now boils down to understanding the number of iterations of the subproblem solver needed in order to satisfy these requirements on ρ and $\{\zeta_k\}$ (or Z).

In the rest of this section, we provide a unified complexity result for solving subproblem (5.8) in Algorithm 4 such that criteria in (5.9) and (5.10) are satisfied for various settings of the objective $\psi(x)$.

Unified method for solving subproblem (5.8) Here we provide a unified complexity analysis for solving subproblem (5.8). In particular, consider the form of the objective $\psi(x) = \mathbb{E}_\xi[\Psi(x, \xi)]$, where ξ is the random input of $\Psi(x, \xi)$ and $\psi(x)$ satisfies the following property:

$$\psi(x) - \psi(y) - \langle \psi'(y), x - y\rangle \leqslant \tfrac{L}{2}\|x - y\|_2^2 + M\|x - y\|_2.$$

Note that, when $M = 0$, function ψ is Lipschitz smooth whereas when $L = 0$, it is nonsmooth. Due to the possible stochastic nature of Ψ, negative lower curvature in (5.6) and the combined smoothness and nonsmoothness property above, we have that ψ can be either smooth or nonsmooth, deterministic or stochastic and convex ($\mu = 0$) or nonconvex ($\mu > 0$). We also assume bounded

second moment stochastic oracle for ψ' when ψ is a stochastic function: For any x, we have an oracle whose output, $\Psi'(x,\xi)$, satisfies $\mathbb{E}_\xi[\Psi'(x,\xi)] = \psi'(x)$ and $\mathbb{E}[\|\Psi'(x,\xi) - \psi'(x)\|_2^2] \leqslant \sigma^2$.

For such a function, we consider an accelerated stochastic approximation algorithm (AC-SA) proposed in [40] for solving the subproblem (5.8) which can be reformulated as $\min_x \psi_k(x) + \mathbf{I}_{\{g_k(x)\leqslant\eta_k\}}(x)$, where $\mathbf{I}$ is the indicator set function. AC-SA algorithm can be applied when $\gamma \geqslant \mu$. In particular, $\psi_k(x) := \psi(x) + \frac{\gamma}{2}\|x - x^{k-1}\|_2^2$ is $(\gamma-\mu)$-strongly convex and $(L+\gamma)$-Lipschitz smooth. Moreover, AC-SA requires computation of a single prox operation of the following form in each iteration:

$$\operatorname*{argmin}_x w^T x + \|x - \bar{x}\|_2^2 + \mathbf{I}_{\{g_k(x)\leqslant\eta_k\}}(x), \tag{5.11}$$

for any $w, \bar{x} \in \mathbb{R}^d$. We show an efficient method for solving this problem at the end of in this section. For now, we look at convergence properties of the AC-SA:

Proposition 5.3.4 *[40] Let x^k be the output of AC-SA algorithm after running T_k iterations for the subproblem* (5.8)*. Then $g_k(x^k) \leqslant \eta_k$ and $\mathbb{E}[\psi_k(x^k) - \psi_k(\bar{x}^k)] \leqslant \frac{2(L+\gamma)}{T_k^2}\|x^{k-1} - \bar{x}^k\|_2^2 + \frac{8(M^2+\sigma^2)}{(\gamma-\mu)T_k}$*

Note that convergence result in Proposition 5.3.4 closely follows the requirement in (5.9). In particular, we should ensure that T_k is big enough such that $\frac{\rho}{2} \leqslant \frac{2(L+\gamma)}{T_k^2}$ and $\zeta_k = \frac{8(M^2+\sigma^2)}{(\gamma-\mu)T_k}$ sum to a constant. Consequently, we have the following corollary:

Corollary 5.3.5 *Let ψ be nonconvex such that it satisfies* (5.6) *with $\mu > 0$. Set $\gamma = 3\mu$ and run AC-SA for $T_k = \max\{2\big(\frac{L}{\mu}+3\big)^{1/2}, K(M+\sigma)\}$ iterations where K is total iterations of Algorithm 4. Then, we obtain that $x^{\hat{k}}$ is an $(\varepsilon_1, \varepsilon_2)$-KKT point of* (5.5)*, where $\hat{k}$ is chosen according to Theorem 5.3.3 and*

$$\varepsilon_1 = \big(\tfrac{3\Delta^0}{2K} + \tfrac{8(M+\sigma)}{\mu K}\big)\max\big\{\tfrac{8(9\mu^2+B^2L_h^2)}{\mu}, \tfrac{2BL_h}{\mu}\big\} + \tfrac{2B(\eta-\eta_0)}{K}, \quad \varepsilon_2 = \tfrac{3\Delta^0}{\mu K} + \tfrac{32(M+\sigma)}{\mu^2 K}$$

Note that Corollary 5.3.5 gives a unified complexity for obtaining KKT point of (5.5) in various settings of nonconvex objective ($\mu > 0$). First, in order to get an ε-KKT point, K must be of $O(1/\varepsilon)$. If the problem is deterministic and smooth then $M = \sigma = 0$. In this case, $T_k = 2(\frac{L}{\mu}+3)^{1/2}$

is a constant. Hence, the total iteration count is $\sum_{k=1}^{K} T_k = O(K)$, implying that total iteration complexity for obtaining an ε-KKT point is of $O(1/\varepsilon)$. For nonsmooth or stochastic cases, M or σ is positive. Hence, $T_k = O(K(M+\sigma))$ implying the total iteration complexity $\sum_{k=1}^{K} T_k = O(K^2)$, which is of $O(1/\varepsilon^2)$. Similar result for the convex case is shown in the appendix.

Efficient projection We conclude this section by formally stating the theorem which provides an efficient oracle for solving the projection problem (5.11). Since $g_k(x) = \lambda\|x\|_1 + \langle v, x\rangle$, the linear form along with ℓ_1 ball breaks the symmetry around origin which is used in existing results on (weighted) ℓ_1-ball projection [31, 52]. Our method involves a careful analysis of Lagrangian duality equations to convert the problem into finding the root of a piecewise linear function. Then a line search method can be employed to find the solution in $O(d \log d)$ time. The formal statement is as follows:

Theorem 5.3.6 *There exists an algorithm that runs in $O(d \log d)$-time and solves the following problem exactly:*

$$\min_{x \in \mathbb{R}^d} \tfrac{1}{2}\|x - v\|_2^2 \quad \textit{subject to} \quad \|x\|_1 + \langle u, x\rangle \leqslant \tau. \tag{5.12}$$

In conclusion, note that (5.11) and (5.12) are equivalent where v in (5.12) can be replaced by $\bar{x} + \frac{1}{2}w$ of (5.11) to get the equivalence of the objective functions of the two problems.

5.4 Numerical Experiments

The goal of this section is to illustrate the empirical performance of LCPP. For simplicity, we will consider the following logistic regression problem:

$$\min_x \ \psi(x) = \tfrac{1}{n}\textstyle\sum_{i=1}^{n} \log(1 + \exp(-b_i a_i^T x)), \quad \text{s.t.} \quad g(x) \leqslant \eta,$$

where $a_i \in \mathbb{R}^d$ is the training sample, $b_i \in \{\pm 1\}$ is the training label, and $g(x)$ is the MCP penalty (see Table 5.2). Details of the testing datasets are summarized in Table 5.3. As we have stated,

LCPP can be equipped with projected first order methods for fast iteration. We compare the efficiency of (spectral) gradient descent [42], Nesterov accelerated gradient and stochastic gradient [112] for solving LCPP subproblem. We find that spectral gradient outperforms the other methods and hence use it in LCPP for the remaining experiment. Due to the space limit, we leave the discussion of this part in appendix. The rest of the section will compare the optimization efficiency of LCPP with the state-of-the-art nonlinear programming solver, and compare the proposed sparse constrained models solved by LCPP with standard convex and nonconvex sparse regularized models. Our first experiment is to compare LCPP with existing optimization library for their

Table 5.3: Dataset description. `mnist` is formulated as a binary problem to classify digit 5 from the other digits. `real-sim` is randomly partitioned into 70% training data and 30% testing data.

Datasets	Training size	Testing size	Dimensionality	Ratio of Nonzeros
`real-sim`	50347	21962	20958	0.25%
`rcv1.binary`	20242	677399	47236	0.16%
`mnist`	60000	10000	784	19.12%
`gisette`	6000	1000	5000	99.10%

optimization efficiency. To the best of our knowledge, DCCP ([99]) is the only open-source package available for the proposed nonconvex constrained problem. While the work [99] has made its code available online, we found that their code had unresolved errors in parsing MCP functions. Therefore, we replicate their setup in our own implementation. DCCP converts the initial problem into a sequence of relatively easier convex problems amenable to CVX ([29]), a convex optimization interface that runs on top of popular optimization libraries. We choose DCCP with MOSEK as the backend as it consistently outperforms DCCP with the default open-source solver SCS.

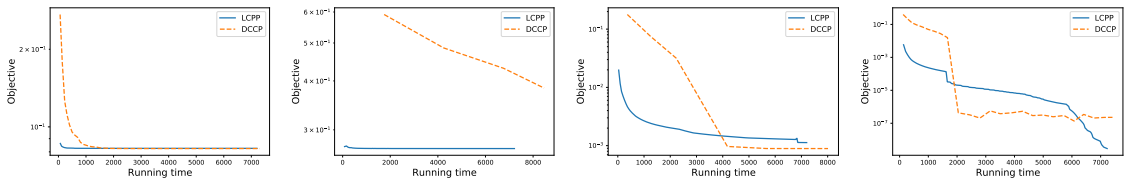


Figure 5.2: Objective value vs. running time (in seconds). Left to right: `mnist` ($\eta = 0.1d$), `real-sim` ($\eta = 0.001d$), `rcv1.binary` ($\eta = 0.05d$) and `gisette` ($\eta = 0.05d$). d stands for the feature dimension.

To fix the parameters, we choose $\gamma = 10^{-5}$ for `gisette` dataset and $\gamma = 10^{-4}$ for the other datasets. For each LCPP subproblem we run gradient descent at most 10 iterations and break when the criterion $\|x^k - x^{k-1}\|/\|x^k\| \leqslant \varepsilon$ is met. We set the number of outer loops as 1000 to run LCPP sufficiently long. We set $\lambda = 2, \theta = 0.25$ in the MCP function. Figure 5.2 plots the convergence performance of LCPP and DCCP, confirming that LCPP is more advantageous over DCCP. Specifically, LCPP outperforms DCCP, sometimes reaching near-optimality even before DCCP finishes the first iteration. This observation can be explained by the fact that LCPP leverages the strengthen of first order methods, for which we can derive efficient projection subroutine. In contrast, DCCP is not scalable to large dataset due to the inefficiency in dealing with large scale linear system arising from the interior point subproblems.

Our next experiment is to compare the performance of nonconvex sparse constrained models, which is then optimized by LCPP, against regularized learning models in the following form:

$$\min_x \ \psi(x) = \tfrac{1}{n}\textstyle\sum_{i=1}^n \log(1 + \exp(-b_i a_i^T x)) + \alpha g(x).$$

In above, $g(x)$ is the sparsity-inducing penalty function. We consider both convex and nonconvex functions, namely Lasso-type penalty $g(x) = \|x\|_1$ and MCP penalty (see Table 5.2). We solve the Lasso problem by Sklearn [87] logistic regression solver and solve the MCP regularized problem by GIST algorithm [42]. For simplicity, both GIST and LCPP set $\lambda = 2$ and $\theta = 5$ in MCP function, and set the maximum iteration number as 2000 for all the algorithms. Then we use a grid of values α for GIST and LASSO, and η for LCPP accordingly, to obtain the classification error under various sparsity levels. Experiment results on average of 10 runs are presented in Figure 5.3. We can clearly see the advantage of our proposed models over Lasso-type estimators. We observe that nonconvex models LCPP and GIST both perform more robustly than Lasso across a wide range of sparsity levels. Lasso models tend to overfit with increasing number of selected features while LCPP is less affected by the feature selection.

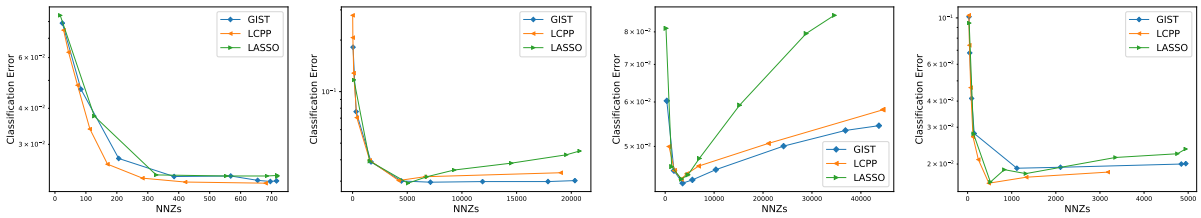


Figure 5.3: Testing error vs number of nonzeros. From left to right: `mnist`, `real-sim`, `rcv1.binary` and `gisette`.

5.5 Auxiliary results

5.5.1 Existence of KKT points

Proposition 5.5.1 *Under Assumption 5.2.3, let $x^0 = \hat{x}$. Then, for any $k \geqslant 1$, we have x^{k-1} is strictly feasible for the k-th subproblem. Moreover, there exists $\bar{x}^k, \bar{y}^k \geqslant 0$ such that $g_k\left(\bar{x}^k\right) \leqslant \eta_k$ and:*

$$\begin{aligned} \partial\psi(\bar{x}^k) + \gamma\left(\bar{x}^k - x^{k-1}\right) + \bar{y}^k\left(\partial g_k(\bar{x}^k)\right) \ni 0 \\ \bar{y}^k\left(g_k\left(\bar{x}^k\right) - \eta_k\right) = 0 \end{aligned} \tag{5.13}$$

Proof. Since x^0 satisfies $g(x^0) \leqslant \eta_0 < \eta_1$ so we have that first subproblem is well defined. We prove the result by induction. First of all, suppose x^{k-1} is strictly feasible for k-th subproblem: $g_k(x^{k-1}) < \eta_k$. Then we note that this problem is also valid and a feasible x^k exists. Hence, algorithm is well-defined. Now, note that

$$g_{k+1}(x^k) = g(x^k) \leqslant g_k(x^k) \leqslant \eta_k < \eta_{k+1}.$$

where first inequality follows due to majorization, second inequality follows due to feasibility of x^k for k-the subproblem and third strict inequality follows due to strictly increasing nature of sequence $\{\eta_k\}$.

Since k-th subproblem has x^{k-1} as strictly feasible point satisfying Slater condition so we obtain existence of $\bar{x}^k$ and $\bar{y}^k \geqslant 0$ satisfying the KKT condition (5.13). □

5.5.2 Proof of Theorem 5.3.1

In order to prove this theorem, we first state the following intermediate result.

Proposition 5.5.2 *Let π_k denotes the randomness of $x^1, x^2, ..., x^{k-1}$. Assume that there exists a $\rho \in [0, \gamma - \mu]$ and a summable nonnegative sequence ζ_k ($\zeta_k \geqslant 0$, $\sum_{k=1}^{\infty} \zeta_k < \infty$) such that*

$$\mathbb{E}\left[\psi_k(x^k) - \psi_k(\bar{x}^k)|\pi_k\right] \leqslant \tfrac{\rho}{2}\|\bar{x}^k - x^{k-1}\|_2^2 + \zeta_k \tag{5.14}$$

Then, under Assumption 5.2.3, we have

1. The sequence $\mathbb{E}[\psi(x^k)]$ is bounded;

2. $\lim_{k\to\infty} \psi(x^k)$ exists a.s.;

3. $\lim_{k\to\infty} \|x^{k-1} - \bar{x}^k\|_=^0$ a.s.;

4. If the whole algorithm is deterministic then $\psi(x^k)$ is bounded. Moreover, if $\zeta_k = 0$, then the sequence $\psi(x^k)$ is monotonically decreasing and convergent.

Proof. Due to the strong convexity of $\psi_k(x)$, we have

$$\psi_k(\bar{x}^k) \leqslant \psi_k(x) - \tfrac{\gamma-\mu}{2}\|\bar{x}^k - x\|_2^2, \tag{5.15}$$

for all x satisfying $g_k(x) \leqslant \eta_k$. Taking $x = x^{k-1}$ and using feasibility of x^{k-1} ($g_k(x^{k-1}) \leqslant \eta_k$) we have

$$\psi(x^{k-1}) \geqslant \psi(\bar{x}^k) + \tfrac{\gamma}{2}\|\bar{x}^k - x^{k-1}\|_2^2 + \tfrac{\gamma-\mu}{2}\|x^{k-1} - \bar{x}^k\|_2^2$$

Together with (5.9) we have

$$\begin{aligned} \zeta_k + \psi(x^{k-1}) \geqslant \mathbb{E}\big[\psi(x^k) + \tfrac{\gamma}{2}\|x^k - x^{k-1}\|_2^2|\pi_k\big] \\ + \tfrac{\gamma-\mu-\rho}{2}\|x^{k-1} - \bar{x}^k\|_2^2. \end{aligned} \tag{5.16}$$

Since $\{\zeta_k\}$ is summable, taking the expectation of π_k and summing up all over all k, we have $\mathbb{E}[\psi(x^k)] \leqslant \psi(x^0) + \sum_{s=1}^{k}\zeta_k < \infty$. Moreover, Applying Supermartingale Theorem 5.5.11 to

(5.16), we have $\lim_{k\to\infty}\psi(x^k)$ exists and $\sum_{k=1}^{\infty}\|x^{k-1}-\bar{x}^k\|_2^2 < \infty$ a.s. Hence we conclude $\lim_{k\to\infty}\|x^{k-1}-\bar{x}^k\|_2 = 0$ a.s. Part 4) can be readily deduced from (5.16). □

Now we are ready to prove Theorem 5.3.1.

For simplicity, we assume the whole sequence generated by Algorithm 4 converges to $\widetilde{x}$. Due to Proposition 5.5.1, there exists a KKT point $(\bar{x}^k, \bar{y}^k)$. The optimality condition yields

$$\psi(x)+\frac{\gamma}{2}\|x-x^{k-1}\|_2^2+\bar{y}^k g_k(x) \geqslant \psi(\bar{x}^k)+\frac{\gamma}{2}\|\bar{x}^k-x^{k-1}\|_2^2+\bar{y}^k g_k(\bar{x}^k), \quad \forall x \tag{5.17}$$

Since $\bar{y}^k$ is bounded, there exists a convergent subsequence $\{i_k\}$ that $\lim_{k\to\infty}\bar{y}^{i_k} = \widetilde{y}$ for some $\widetilde{y} \geqslant 0$. Let us take $k\to\infty$ in (5.17). In view of Proposition 5.5.2, Part 3, we have $\lim_{k\to\infty}\bar{x}^{i_k} = \lim_{k\to\infty}x^{i_k-1} = \widetilde{x}$ almost surely. Then $\lim_{k\to\infty}h(x^{i_k-1}) = h(\widetilde{x})$ and $\lim_{k\to\infty}\nabla h(x^{i_k-1}) = \nabla h(\widetilde{x})$ a.s. due to the continuity of $h(x)$ and $\nabla h(x)$, respectively. Then we have

$$\psi(x)+\frac{\gamma}{2}\|x-\widetilde{x}\|_2^2+\widetilde{y}\left[\lambda\|x\|_1-h(\widetilde{x})-\langle\nabla h(\widetilde{x}),x-\widetilde{x}\rangle\right] \geqslant \psi(\widetilde{x})+\widetilde{y}g(\widetilde{x}), \quad a.s.$$

implying that $\widetilde{x}$ minimizes the loss function $\psi(x)+\frac{\gamma}{2}\|x-\widetilde{x}\|_2^2+\widetilde{y}\left[\lambda\|x\|_1^- h(\widetilde{x})-\langle\nabla h(\widetilde{x}),x-\widetilde{x}\rangle\right]$. Due to the first order optimality condition, we conclude $0\in\partial\psi(\widetilde{x})+\widetilde{y}\partial g(\widetilde{x})$, a.s.

Moreover, using the complementary slackness, we have $0=\bar{y}^{i_k}\left(g_{i_k}\left(\bar{x}^{i_k}\right)-\eta_{i_k}\right)$. Taking the limit of $k\to\infty$ and noticing that $\lim_{k\to\infty}\eta_{i_k}=\eta$, we have $0=\widetilde{y}\left(g\left(\widetilde{x}\right)-\eta\right)$ a.s . As a result, we conclude that $(\widetilde{x},\widetilde{y})$ is a KKT point of problem (5.5), a.s.

5.5.3 Proof of Theorem 5.3.2

From KKT condition of (5.13), $\bar{x}^k$ is the optimal solution of the problem $\min_{x\in\mathbb{R}^d}\ \psi_k(x)+\bar{y}^k\left(g_k(x)-\eta_k\right)$. Therefore, for any $x\in\mathbb{R}^d$, we have

$$\psi_k(x)+\bar{y}^k g_k(x) \geqslant \psi_k(\bar{x}^k)+\bar{y}^k g_k(\bar{x}^k) \tag{5.18}$$

We prove that $\{\bar{y}^k\}$ is bounded a.s. by contradiction. If $\left\{\bar{y}^k\right\}$ has unbounded subsequence with

positive probability, then conditioned under that event, there exists a subsequence $\{i_k\}$ such that $\bar{y}^{i_k} \to \infty$. Let us divide both sides of (5.18) by $\bar{y}^k$ and expand g_k by its definition. After placing $k = i_k$, we have for all x

$$\begin{aligned} &\tfrac{1}{\bar{y}^{i_k}}\psi_{i_k}(x) + \lambda\|x\|_1 - \nabla h(x^{i_k-1})^T x \\ &\geqslant \tfrac{1}{\bar{y}^{i_k}}\psi_{i_k}(\bar{x}^{i_k}) + \lambda\|\bar{x}^{i_k}\|_1 - \nabla h(x^{i_k-1})^T \bar{x}^{i_k}. \end{aligned} \tag{5.19}$$

Let $\widetilde{x}$ be any limiting point a.s. of the sequence $\{x^{i_k-1}\}$. By the statement of the theorem, we know that it exists and satisfies MFCQ assumption. Passing to some subsequence if necessary, we have $\lim_{k\to\infty} x^{i_k-1} = \widetilde{x}$ a.s. Using Proposition 5.5.2 Part 3, we have $\lim_{k\to\infty} \bar{x}^{i_k} = \widetilde{x}$ a.s. Moreover, using Proposition 5.5.2 Part 2, we have $\lim_{k\to\infty} \psi(\bar{x}^{i_k})$ exists a.s. This implies $\lim_{k\to\infty} \frac{1}{\bar{y}^{i_k}}\psi_{i_k}(\bar{x}^{i_k}) = 0$ a.s.

Taking $k \to \infty$, since $\psi_{i_k}(x)$ is bounded a.s. (due to existence of $\widetilde{x}$ a.s.), we have $\lim_{k\to\infty} \frac{1}{\bar{y}^{i_k}}\psi_{i_k}(x) = 0$. From Lipschitz continuity of l_1 norm and $\nabla h(x)$, we have $\lim_{k\to\infty} \lambda\|\bar{x}^{i_k}\|_1 = \lambda\|\widetilde{x}\|_1$ a.s., and $\lim_{k\to\infty} \nabla h(x^{i_k-1}) = \nabla h(\widetilde{x})$ a.s., respectively. It then follows from (5.19) that for all x, we have $\lambda\|x\|_1 - \langle \nabla h(\widetilde{x}), x\rangle \geqslant \lambda\|\widetilde{x}\|_1 - \langle \nabla h(\widetilde{x}), \widetilde{x}\rangle$. In other words, we have

$$\mathbf{0} \in \partial\lambda\|\widetilde{x}\|_1 - \nabla h(\widetilde{x}) = \partial g(\widetilde{x}), a.s. \tag{5.20}$$

Moreover, due to complementary slackness and $\bar{y}^{i_k} > 0$, the equality $g_{i_k}(\bar{x}^{i_k}) = \eta_{i_k}$ holds. Hence, in the limit, we have the constraint $g(\widetilde{x}) = \eta$ active a.s. Under MFCQ, there exists z such that $\max_{v\in\partial g(\widetilde{x})} z^T v < 0$. However, from (5.20) we have $0 = z^T\mathbf{0}$ since $\mathbf{0} \in \partial g(\widetilde{x})$, leading to a contradiction to the event that $\{\bar{y}^k\}$ contained unbounded sequence with positive probability. Hence, $\bar{y}$ is bounded a.s.

5.5.4 Explicit and specialized bounds on the dual

Here, we discuss some of the results for explicit bounds on the dual. In particular, we focus on the SCAD and MCP case. Similar results can be extended for Exp and $\ell_p, p < 0$ case since these

function follows two key properties (as we will see later in the proofs):

1. $|\nabla h(x)| \leqslant \lambda$ for all x for each of these functions.

2. They remain bounded below a constant. See Figure 5.1.

We exploit these two structural properties of these sparse constraints to obtain specialized and explicit bounds on the optimal dual of problem 5.5. The following lemma is in order.

Lemma 5.5.3 *Let $h : \mathbb{R} \to \mathbb{R}$ be the the convex function which satisfies $|\nabla h(x)| \leqslant \lambda$ for all $x \in \mathbb{R}$. Then the minimum value of $\overline{g}(x; \overline{x}) : \mathbb{R} \to \mathbb{R}$ defined as $\overline{g}(x; \overline{x}) := \lambda|x| - h(\overline{x}) - \langle \nabla h(\overline{x}), x - \overline{x} \rangle$ is achieved at 0 for all $\overline{x} \in \mathbb{R}$.*

Proof. Note that $\overline{g}$ is a convex function for any $\overline{x} \in \mathbb{R}$. So by first order optimality condition, if $\widehat{x}$ is the minimizer of $\overline{g}$ then $0 \in \partial \overline{g}(\widehat{x}; \overline{x})$. This implies

$$\lambda \partial |\widehat{x}| - \nabla h(\overline{x}) \ni 0.$$

Note that $\widehat{x} = 0$ satisfies this condition since in that case $\lambda \partial |\widehat{x}| = [-\lambda, \lambda]$. And due to assumption on h, we have $\nabla h(\overline{x}) \in [-\lambda, \lambda]$. Hence $\widehat{x} = 0$ is always the minimizer. ▫

Now note that $h_{\lambda,\theta}$ functions defined for our examples, such as SCAD or MCP. satisfy the assumption of bounded gradients in Lemma 5.5.3. Now we use this simple result to show that $\mathbf{0}$ is the most feasible solution for each of the subproblem (5.8) generated in Algorithm 4 and hence we can give an explicit bound for the optimal dual value for each subproblem.

Lemma 5.5.4 *Suppose all assumptions in Lemma 5.5.3 are satisfied. Then we have for any $k \geqslant 1$,*

$$\overline{y}^k \leqslant \frac{\psi_k(\mathbf{0}) - \psi_k(\bar{x}^k)}{\eta_k - g(x^{k-1}) + \sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}|}. \tag{5.21}$$

Proof. Note that $g_k(x) = \sum_{i=1}^d \overline{g}(x_i; x_i^{k-1})$ where $\overline{g}$ is defined in Lemma 5.5.3. Since assumptions of Lemma 5.5.3 hold, so we have that each individual $\overline{g}$ is minimized at $x_i = 0$. Hence $g_k(\mathbf{0})$ is the

minimum value of g_k. In view of Proposition 5.5.1, we have that x_{k-1} is strictly feasible solution with respect to constraint $g_k(x) \leqslant \eta_k$ implying $g_k(x^{k-1}) - \eta_k < 0$. Hence, we have

$$\begin{aligned}
&\eta_k - g_k(\mathbf{0}) \\
&= \eta_k - \left[\lambda\|\mathbf{0}\|_1 - \textstyle\sum_{i=1}^d \{h(x_i^{k-1}) + \nabla h(x_i^{k-1})(0 - x_i^{k-1})\}\right] \\
&= \eta_k + \textstyle\sum_{i=1}^d h(x_i^{k-1}) - \sum_{i=1}^d \nabla h(x_i^{k-1}) x_i^{k-1} \\
&= \eta_k - g(x^{k-1}) + [g(x^{k-1}) + h(x^{k-1})] - \textstyle\sum_{i=1}^d \nabla h(x_i^{k-1}) x_i^{k-1} \\
&\geqslant \eta_k - g(x^{k-1}) + \lambda\|x^{k-1}\|_1 - \textstyle\sum_{i=1}^d |\nabla h(x_i^{k-1})||x_i^{k-1}| \\
&= \eta_k - g(x^{k-1}) + \textstyle\sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}| \\
&> 0.
\end{aligned}$$

Here, last strict inequality follows due to the fact that $\lambda \geqslant |\nabla h(x_i^{k-1})|$ and $\eta_k > g(x^{k-1})$. Then, we have, optimal dual $\bar{y}^k$ satisfies for all x:

$$\begin{aligned}
\psi_k(\bar{x}^k) &\leqslant \psi_k(x) + \bar{y}^k(g_k(x) - \eta_k) \\
\Rightarrow \psi_k(\bar{x}^k) &\leqslant \psi_k(\mathbf{0}) + \bar{y}^k(g_k(\mathbf{0}) - \eta_k) \\
\Rightarrow \bar{y}^k &\leqslant \tfrac{\psi_k(\mathbf{0}) - \psi_k(\bar{x}^k)}{\eta_k - g_k(\mathbf{0})} \\
&\leqslant \frac{\psi_k(\mathbf{0}) - \psi_k(\bar{x}^k)}{\eta_k - g(x^{k-1}) + \sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}|},
\end{aligned}$$

where third inequality follows due to the fact that $\eta_k - g_k(\mathbf{0}) > 0$ Hence, we conclude the proof. ▫

Note that the bound in (5.21) depends on x^{k-1} which can not be controlled, especially in the stochastic cases. In order to show a bound on $\bar{y}^k$ irrespective of x^{k-1}, we must lower bound the denominator in (5.21) for all possible values of x^{k-1}. To accomplish this goal, we show the following two theorems in which we lower bound the term $\sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}|$. Each of these theorem is a specialized result for SCAD and MCP function, respectively.

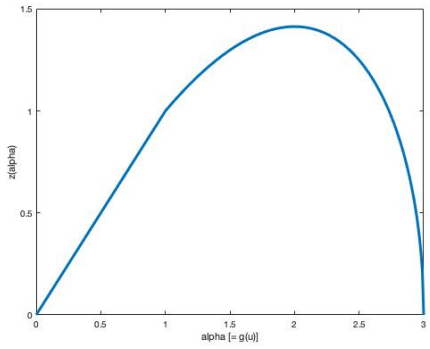


Figure 5.4: Plot of $z(\gamma)$ for SCAD function where $\lambda = 1,\ \theta = 5.\ z : [0,3] \to \mathbb{R}_{\geqslant 0}$ where $z(0) = z(3) = 0$ otherwise z is strictly positive.

Theorem 5.5.5 *Let g be the SCAD function and $x \in \mathbb{R}^d$ such that $g(x) = \alpha$. Also, let $\gamma = \alpha - \beta\frac{\lambda^2(\theta+1)}{2}$ where β is the largest nonnegative integer such that $\gamma \geqslant 0$. Then, $\sum_{i=1}^{d}(\lambda - |\nabla h(x_i)|)|x_i| \geqslant z(\gamma)$ where $z : [0, \frac{\lambda^2(\theta+1)}{2}] \to \mathbb{R}_{\geqslant 0}$ is the function defined as*

$$z(\gamma) := \begin{cases} \gamma & \text{if } 0 \leqslant \gamma \leqslant \lambda^2 \\ \frac{\gamma}{\lambda}\sqrt{\frac{2}{\theta-1}}\sqrt{\frac{\lambda^2(\theta+1)}{2} - \gamma} & \text{if } \lambda^2 < \gamma \leqslant \frac{\lambda^2(\theta+1)}{2} \end{cases}.$$

Theorem 5.5.6 *Let g be the MCP function and $x \in \mathbb{R}^d$ be such that $g(x) = \alpha$. Also let $\gamma = \alpha - \beta\frac{\lambda^2\theta}{2}$ where β is the largest nonnegative integer such that $\gamma \geqslant 0$. Then $\sum_{i=1}^{d}(\lambda - |\nabla h(x_i)|)|x_i| \geqslant z(\gamma)$ where $z : [0, \frac{\lambda^2\theta}{2}] \to \mathbb{R}_{\geqslant 0}$ is the function defined as $z(\gamma) := \gamma\sqrt{1 - \frac{2\gamma}{\theta\lambda^2}}$.*

Note that Theorem 5.5.5 states that lower bound $z(\gamma) = 0$ when $\gamma = 0$ or $\frac{\lambda^2(\theta+1)}{2}$. In essence, when α is exact integral multiple of $\frac{\lambda^2(\theta+1)}{2}$ then lower bound turn out to be zero. However, for all other values of α, the corresponding $z(\gamma)$ is strictly positive. This can be seen from the graph of $z(\gamma)$ below. Similar claims can be made with respect to MCP in Theorem 5.5.6.

Now we are ready to show a bound on $\bar{y}^k$ irrespective of x^{k-1}. We give a specific routine to choose the values of η_k such that we can obtain a provable bound on the denominator in (5.21) hence obtaining an upper bound on the $\bar{y}^k$ for all k irrespective of x^{k-1}.

Proposition 5.5.7 *Let g be the SCAD function and $\eta = \beta\frac{\lambda^2(\theta+1)}{2} + \widetilde{\eta}$ where β be the largest nonnegative integer such that $\widetilde{\eta} \geqslant 0$. Then, for properly selected η_0, we have that $\eta_k - g(x^{k-1}) +$*

$\sum_{i=1}^{d}(\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}| \geqslant \min\{\lambda^2, \frac{z(\widetilde{\eta})}{2}\}$.

We note that very similar proposition for MCP can be proved based on Theorem 5.5.6. We skip that discussion in order to avoid repetition.

Connection to MFCQ In this section, we show the connection of MFCQ assumption in Theorem 5.3.2 with the bound in Theorem 5.5.5.

Note that for the boundary points of the set $g(x) \leqslant \eta_1$ where $\eta_1 = \frac{\lambda^2(\theta+1)}{2}$ then the lower bound $z(\eta_1) = 0$. In fact, carefully following the proof of Theorem 5.5.5, we can identify that the lower bound is tight for x's such that one of the coordinate x_i satisfy $|x_i| \geqslant \lambda\theta$ and all other coordinates are 0. In this case, we see that such points do not satisfy MFCQ. At such points, we don't have any strictly feasible directions required by MFCQ assumption. This can be easily visualized in the leftmost figure in Figure 5.5. Note that $\lambda\theta = 5$ and for any $|x| \geqslant 5$, the feasible region is merely the axis and hence there is no strict feasible direction. This implies MFCQ indeed fails at these points.

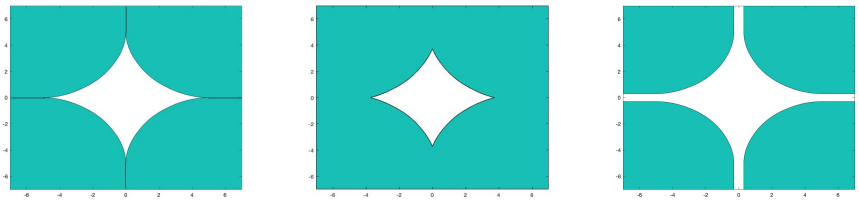


Figure 5.5: All figures are plotted for $\lambda = 1$ and $\theta = 5$. From left to right: $\eta_1 = 3, \eta_2 = 2.8$ and $\eta_3 = 3.2$. Then $\eta_1 = \frac{\lambda^2(\theta+1)}{2} = 3$. In first figure, we see that for $|x| \geqslant 5$, the MFCQ assumption is violated since only x-axis is feasible. Similar observation holds for y-axis as well. However, in second and third figure such claims are no longer valid.

For $g(x) = \eta_2 < \eta_1$ the lower bound $z(\eta_2)$ is nonzero and same holds for $g(x) = \eta_3 > \eta_1$. Indeed, we see that for such cases, the points not satisfying MFCQ in case of η_1 vanish. This can be observed in second and third figure in Figure 5.5. For the case of η_2 in part (b), these points become infeasible and for the case of η_3 in part (c), they are no longer boundary points.

Looking back at MFCQ from the result of Theorem 5.5.5, we can see that how close η is to $\frac{\lambda^2(\theta+1)}{2}$ shows how 'close' the problem is for violating MFCQ. Moreover, the lower bound $z(\cdot)$ on

the denominator of (5.21) shows how quickly the dual will explode as the problem setting gets closer to violating MFCQ.

We complete this discussion by showing the proof of Theorem 5.5.5 and Theorem 5.5.6. We also note that similar theorems can be proved for $\ell_p, p < 0$ and Exp function in Table 5.2.

Proof of Theorem 5.5.5

First, we show a lower bound for one-dimensional function and then extend it to higher dimensions. Suppose $u \in \mathbb{R}$ be such that $g(u) = \alpha$. Note that since g is SCAD function so α must lie in the set $[0, \frac{\lambda^2(\theta+1)}{2}]$. Key to our analysis is the lower bound on $(\lambda - |\nabla h(u)|)|u|$ as a function of α. Note that since

$$g(u) = \alpha \Rightarrow \lambda|u| \geqslant \alpha \Rightarrow |u| \geqslant \tfrac{\alpha}{\lambda}. \tag{5.22}$$

Also note that for all $|u| \leqslant \lambda$, we have $g(u) = \lambda|u|$ and $\nabla h(u) = 0$ which implies $\nabla h(u) = 0$ for all $g(u) = \alpha \leqslant \lambda^2$. Hence, using this relation along with (5.22), we obtain

$$(\lambda - |\nabla h(u)|)|u| = \lambda|u| \geqslant \alpha \qquad \text{if } 0 \leqslant \alpha \leqslant \lambda^2. \tag{5.23}$$

We note that $|\nabla h(u)| = \lambda$ for all $u \geqslant \lambda\theta$ and $g(u) = \alpha = \frac{\lambda^2(\theta+1)}{2}$ for all $u \geqslant \lambda\theta$. Hence,

$$(\lambda - |\nabla h(u)|)|u| = 0 \qquad \text{if } \alpha = \tfrac{\lambda^2(\theta+1)}{2}. \tag{5.24}$$

Now we design a lower bound when $\alpha \in (\lambda^2, \frac{\lambda^2(\theta+1)}{2})$. For such values of α, we have

$$\begin{aligned}
g(u) =& \lambda|u| - \tfrac{(|u|-\lambda)^2}{2(\theta-1)} = \alpha \\
\Rightarrow& u^2 - 2\lambda\theta|u| + \lambda^2 + 2\alpha(\theta-1) = 0 \\
\Rightarrow& |u| = \lambda\theta - \sqrt{2(\theta-1)\big[\tfrac{\lambda^2(\theta+1)}{2} - \alpha\big]} \\
\Rightarrow& |\nabla h(u)| = \tfrac{|u|-\lambda}{\theta-1} = \lambda - \sqrt{\tfrac{2}{\theta-1}}\sqrt{\tfrac{\lambda^2(\theta+1)}{2} - \alpha} \\
\Rightarrow& \lambda - |\nabla h(u)| = \sqrt{\tfrac{2}{\theta-1}}\sqrt{\tfrac{\lambda^2(\theta+1)}{2} - \alpha}.
\end{aligned}$$

Then, above relation along with (5.22), we have $(\lambda - |\nabla h(u)|)|u| \geqslant \sqrt{\frac{2}{\theta-1}}\frac{\alpha}{\lambda}\sqrt{\frac{\lambda^2(\theta+1)}{2} - \alpha}$ for all $\alpha \in (\lambda^2, \frac{\lambda^2(\theta+1)}{2})$. Using this relation along with (5.23), (5.24) and noting the definition of function $z(\cdot)$, we obtain a lower bound $(\lambda - |\nabla h(u)|)|u| \geqslant z(\alpha)$ where $\alpha = g(u)$.

Now note that for general high-dimensional $x \in \mathbb{R}^d$, we have $g(x) = \sum_{i=1}^{d} g(x_i) = \alpha$. Then $\alpha \in [0, \frac{d\lambda^2(\theta+1)}{2}]$. Since each individual $g(x_i) \geqslant 0$, we can think of α as a budget such that sum of $g(x_i)$ must equal α. In order to minimize the lower bound on $(\lambda - |\nabla h(x_i)|)|x_i|$, we should exhaust the largest budget from $\sum_{i=1}^{d} g(x_i) = \alpha$ while maintaining the lowest possible value of the lower bound on $(\lambda - |\nabla h(x_i)|)|x_i|$. This clearly holds by setting $|x_i|$ such that $g(x_i) = \frac{\lambda^2(\theta+1)}{2}$. This can be clearly observed in the figure below.

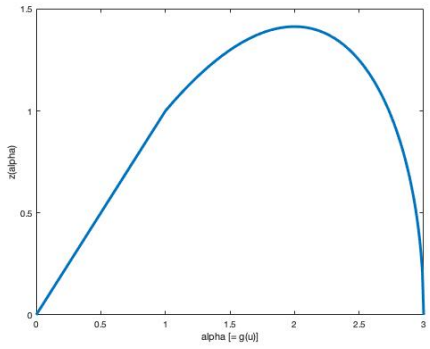


Figure 5.6: Plot of function $z(\alpha)$ on y-axis and α on x-axis for $\lambda = 1,\ \theta = 5$. The largest possible value $g(u)$ is $\frac{\lambda^2(\theta+1)}{2} = 3$ is achieved for $u \geqslant \lambda\theta = 5$ and lower bound $z(3) = 0$. Hence, setting $u \geqslant \lambda\theta$ maximizes the $g(u)$ and minimizes $z(\alpha) = z(g(u))$.

Hence, if $\alpha \in \left[\beta\frac{\lambda^2(\theta+1)}{2}, (\beta+1)\frac{\lambda^2(\theta+1)}{2}\right)$ for some nonnegative integer β, then we should set β coordinates of x satisfying $|x_i| \geqslant \lambda\theta$ in order to exhaust the maximum possible budget, $\frac{\lambda^2(\theta+1)}{2}$, from α and still keep the value of the lower bound on $(\lambda - |\nabla h(u)|)|u|$ as 0. Hence, noting the definition of γ, the problem reduces to $\sum_i g(x_i) = \gamma$ where summation is taken over remaining coordinates of x and $\gamma \in \left[0, \frac{\lambda^2(\theta+1)}{2}\right)$.

Lets recall from the analysis in 1-D case that if $g(x_i) = \alpha_i$ then $(\lambda - |\nabla h(x_i)|)|x_i| \geqslant z(\alpha_i)$ so we obtain the lower bound $\sum_i z(\alpha_i)$ while α_i's satisfy the relation $\sum_i \alpha_i = \gamma$. Moreover, $z : [0, \frac{\lambda^2(\theta+1)}{2}] \to \mathbb{R}_{\geqslant 0}$ is a concave function with $z(0) = 0$. Then we show that z is a subadditive function. Using Jensen's inequality, for all $t \in [0, 1]$, we have $z(tx+(1-t)y) \geqslant tz(x)+(1-t)z(y)$.

Using $y = 0$ and the fact that $z(0) = 0$, we have $z(tx) \geqslant tz(x)$ for any $t \in [0, 1]$. Now using this relation along with $t = \frac{x}{x+y} \in [0, 1]$ (for $x, y \geqslant 0$) we have

$$z(x) = z(t(x + y)) \geqslant tz(x + y).$$
$$z(y) = z((1 - t)(x + y)) \geqslant (1 - t)z(x + y).$$

Adding the two relations, we obtain $z(x) + z(y) \geqslant z(x + y)$. Hence, z is a subadditive function. Since $\sum_i \alpha_i = \gamma$ then the we have $\sum_i z(\alpha_i) \geqslant z(\sum_i \alpha_i) = z(\gamma)$. This bound is indeed achieved when we set one of $\alpha_i = \gamma$ and rest to 0. Hence, we conclude the proof.

Proof of Theorem 5.5.6

As before, we proceed by assuming 1-D case, i.e., $u \in \mathbb{R}$ and $g(u) = \alpha$ and then extend it to general d-dimensional setting. Then, $\alpha \in [0, \frac{\lambda^2\theta}{2}]$. Then, we write function $(\lambda - |\nabla h(u)|)|u|$ in term of α. Note that

$$\begin{aligned} g(u) = \lambda|u| - \tfrac{u^2}{2\theta} &= \alpha \\ \Rightarrow |u| &= \theta\lambda\big(1 - \sqrt{1 - \tfrac{2\alpha}{\theta\lambda^2}}\big) \\ \Rightarrow |\nabla h(u)| = \tfrac{|u|}{\theta} &= \lambda\big(1 - \sqrt{1 - \tfrac{2\alpha}{\theta\lambda^2}}\big) \\ \Rightarrow \lambda - |\nabla h(u)| &= \lambda\sqrt{1 - \tfrac{2\alpha}{\theta\lambda^2}} \end{aligned}$$

Moreover, we also have (5.22). Then, noting the definition of $z(\cdot)$, we obtain that $(\lambda - |\nabla h(u)|)|u| \geqslant z(\alpha)$.

For high dimensional $x \in \mathbb{R}^d$, we use similar arguments as in the proof of theorem 5.5.5. In particular, we set β coordinates x satisfying $|x_i| \geqslant \lambda\theta$ which exhausts the maximum possible budget $\frac{\lambda^2\theta}{2}$ from α and still keeps the value of the lower bound on $(\lambda - |\nabla h(x_i)|)|x_i|$ as 0. Finally, we reduce the problem to $\sum_i g(x_i) = \sum_i \alpha_i = \gamma$ and lower bound is $\sum_i z(\alpha_i)$. As in the previous case, z is concave function on nonnegative domain with $z(0) = 0$ hence it must be subadditive. So

we obtain that $\sum_i z(\alpha_i) \geqslant z(\sum_i \alpha_i) = z(\gamma)$. Hence, we conclude the proof.

Proof of Proposition 5.5.7

We note that $\eta = \beta\frac{\lambda^2(\theta+1)}{2} + \widetilde{\eta}$, where β is the largest nonnegative integer such that $\widetilde{\eta} \geqslant 0$. Clearly $\widetilde{\eta} \in \left[0, \frac{\lambda^2(\theta+1)}{2}\right)$. Now, we divide our analysis in two cases:

Case 1: Suppose $\widetilde{\eta} \leqslant \lambda^2$. Then we define η_0 for Algorithm 4 as $\eta_0 = \beta\frac{\lambda^2(\theta+1)}{2} + \frac{\widetilde{\eta}}{2}$.

Now, if $g(x^{k-1}) \leqslant \beta\frac{\lambda^2(\theta+1)}{2}$ then we have that $\eta_{k-1} - g(x^{k-1}) \geqslant \eta_0 - g(x^{k-1}) \geqslant \frac{\widetilde{\eta}}{2}$. In this case, we obtain that denominator of (5.21) is at least $\frac{\widetilde{\eta}}{2}$.

In other case, suppose that $g(x^{k-1}) > \beta\frac{\lambda^2(\theta+1)}{2}$. We also note that $g(x^{k-1}) \leqslant g_{k-1}(x^{k-1}) \leqslant \eta_{k-1} \leqslant \eta$. Hence, we obtain $g(x^{k-1}) \leqslant \eta = \beta\frac{\lambda^2(\theta+1)}{2} + \widetilde{\eta}$. This implies $\widetilde{g}(x^{k-1}) := g(x^{k-1}) - \beta\frac{\lambda^2(\theta+1)}{2} \in [0, \lambda^2]$. Then, using Theorem 5.5.5, we obtain that $\sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}| \geqslant z(\widetilde{g}(x^{k-1})) = \widetilde{g}(x^{k-1})$. Using this relation, we obtain that $\eta_{k-1} - g(x^{k-1}) + \sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}| \geqslant \eta_{k-1} - g(x^{k-1}) + \widetilde{g}(x^{k-1}) = \eta_{k-1} - \beta\frac{\lambda^2(\theta+1)}{2} = \widetilde{\eta}_{k-1} \geqslant \frac{\widetilde{\eta}}{2}$.

So, when $\widetilde{\eta} \leqslant \lambda^2$, we obtain that the denominator in (5.21) is at least $\eta_k - \eta_{k-1} + \frac{z(\widetilde{\eta})}{2} = \delta_k + \frac{z(\widetilde{\eta})}{2} \geqslant \frac{z(\widetilde{\eta})}{2}$.

Case 2: Now, we look at the second case where $\widetilde{\eta} > \lambda^2$. In this case, we define $\eta_0 = \beta\frac{\lambda^2(\theta+1)}{2} + \min\{\lambda^2, z(\widetilde{\eta})\}$. Then, we again note that $g(x^{k-1}) \leqslant \beta\frac{\lambda^2(\theta+1)}{2}$ implies $\eta_{k-1} - g(x^{k-1}) \geqslant \widetilde{\eta}_{k-1} \geqslant \widetilde{\eta}_0$. In other case, we assume that $g(x^{k-1}) \in [\beta\frac{\lambda^2(\theta+1)}{2}, \beta\frac{\lambda^2(\theta+1)}{2} + \lambda^2]$, then again using Theorem 5.5.5, we obtain $\sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}| \geqslant z(\widetilde{g}(x^{k-1})) = \widetilde{g}(x^{k-1})$. This implies $\eta_{k-1} - g(x^{k-1}) + \sum_{i=1}^d (\lambda - |\nabla h(x_i^{k-1})|)|x_i^{k-1}| \geqslant \eta_{k-1} - \beta\frac{\lambda^2(\theta+1)}{2} = \widetilde{\eta}_{k-1} \geqslant \widetilde{\eta}_0$.

Finally, $g(x^{k-1}) > \beta\frac{\lambda^2(\theta+1)}{2} + \lambda^2$ then $\widetilde{g}(x^{k-1}) \in (\lambda^2, \widetilde{\eta})$ then due to concavity of z, we obtain that $z(\widetilde{g}(x^{k-1})) \geqslant \min\{\lambda^2, z(\widetilde{\eta})\} = \widetilde{\eta}_0$.

Hence, combining the bounds in both cases, we obtain that denominator in (5.21) is always bounded below by $\min\{\lambda^2, \frac{z(\eta)}{2}\}$.

5.5.5 Proof of Theorem 5.3.3

As in the previous case, we show an important recursive property of iterates. We first state the theorem again:

Theorem 5.5.8 *Suppose Assumption 5.2.3, 5.3.1 hold such that $\delta_k = \frac{\eta-\eta_0}{k(k+1)}$ for all $k \geqslant 1$. Let π_k denote the randomness of $x^1, \ldots, x^{k-1}$. Suppose for k-th subproblem* (5.8)*, the solution x^k satisfies*

$$\mathbb{E}[\psi_k(x^k) - \psi_k(\bar{x}^k)|\pi_k] \leqslant \tfrac{\rho}{2}\|x^{k-1} - \bar{x}^k\|_2^2 + \zeta_k,$$
$$g_k(x^k) \leqslant \eta_k$$

where ρ lies in the interval $[0, \gamma - \mu]$ and $\{\zeta_k\}$ is a sequence of nonnegative numbers. If $\hat{k}$ is chosen uniformly randomly from $\left\lfloor \frac{K+1}{2} \right\rfloor$ to K then corresponding to $x^{\hat{k}}$, there exists pair $(\bar{x}^{\hat{k}}, \bar{y}^{\hat{k}})$ satisfying

$$\mathbb{E}_{\hat{k}}\big[\operatorname{dist}\big(\partial_x \mathcal{L}(\bar{x}^{\hat{k}}, \bar{y}^{\hat{k}}), 0\big)^2\big] \leqslant \tfrac{8(\gamma^2 + B^2 L_h^2)}{K(\gamma-\mu-\rho)}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2Z_1\big),$$
$$\mathbb{E}_{\hat{k}}\big[\bar{y}^{\hat{k}}\big|g(\bar{x}^{\hat{k}}) - \eta\big|\big] \leqslant \tfrac{2BL_h}{K(\gamma-\mu-\rho)}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2Z_1\big) + \tfrac{2B(\eta-\eta_0)}{K},$$
$$\mathbb{E}_{\hat{k}}\|x^{\hat{k}} - \bar{x}^{\hat{k}}\|_2^2 \leqslant \tfrac{4\rho(\gamma-\mu+\rho)}{K(\gamma-\mu)^2(\gamma-\mu-\rho)}\Delta^0 + \tfrac{8Z_1}{K(\gamma-\mu-\rho)},$$

where, $\Delta^0 := \psi(x^0) - \psi(x^*)$ and $Z_1 := \sum_{k=1}^K \zeta_k$.

We first prove the following important relationship on the sum of squares of distances of the iterates.

Proposition 5.5.9 *Let requirements of Theorem 5.3.3 hold. Then for any $s \geqslant 2$, we have*

$$\mathbb{E}[\textstyle\sum_{k=s}^K \|x^{k-1} - \bar{x}^k\|_2^2 | \pi_{s-1}] \leqslant \tfrac{2(A_s + Z_s)}{\gamma-\mu-\rho}, \tag{5.25}$$
$$\mathbb{E}[\textstyle\sum_{k=s}^K \|x^{k} - \bar{x}^k\|_2^2 | \pi_{s-1}] \leqslant \tfrac{2\rho A_s}{(\gamma-\mu)(\gamma-\mu-\rho)} + \tfrac{2Z_s}{\gamma-\mu-\rho} \tag{5.26}$$

where $A_s = \frac{\gamma-\mu+\rho}{\gamma-\mu}\left[\psi(x^{s-2}) - \psi(x^*)\right]$ and $Z_s = \sum_{k=s-1}^K \zeta_k$.

Proof. Note that since for all $k \geqslant 1$ we have feasibility of x^k for k-th subproblem (due to (5.10)), then in view of Proposition 5.5.1, we have that x^{k-1} is strictly feasible for the k-th subproblem. Consequently, using strong convexity of ψ_k and optimality of $\bar{x}^k$, we have $\frac{\gamma-\mu}{2}\|x^{k-1} - \bar{x}^k\|_2^2 \leqslant \psi_k(x^{k-1}) - \psi_k(\bar{x}^k)$. Therefore, taking expectation conditioned on π_{k-1} ob both sides of the above relation, we obtain

$$\begin{aligned}\tfrac{\gamma-\mu}{2}\mathbb{E}[\|x^{k-1} - \bar{x}^k\|_2^2|\pi_{k-1}] &\leqslant \mathbb{E}[\psi_k(x^{k-1}) - \psi_k(\bar{x}^k)|\pi_{k-1}] \\ &\leqslant \mathbb{E}[\psi_{k-1}(x^{k-1}) - \psi_k(\bar{x}^k)|\pi_{k-1}] \\ &\leqslant \psi_{k-1}(\bar{x}^{k-1}) - \mathbb{E}[\psi_k(\bar{x}^k)|\pi_{k-1}] + \tfrac{\rho}{2}\|x^{k-2} - \bar{x}^{k-1}\|_2^2 + \zeta_{k-1}\end{aligned}$$

where second inequality follows from $\psi_k(x^{k-1}) = \psi(x^{k-1}) \leqslant \psi_{k-1}(x^{k-1})$ and third inequality follows from (5.9). Placing the definition of $\psi_k(\cdot)$ in above relation, we have

$$\tfrac{2\gamma-\mu}{2}\mathbb{E}[\|x^{k-1} - \bar{x}^k\|_2^2|\pi_{k-1}] \leqslant \psi(\bar{x}^{k-1}) - \mathbb{E}[\psi(\bar{x}^k)|\pi_{k-1}] + \tfrac{\gamma+\rho}{2}\|x^{k-2} - \bar{x}^{k-1}\|_2^2 + \zeta_{k-1}.$$

Summing up over $k = s, s+1, \ldots, K$ and taking expectation conditioned on π_{s-1}, we have

$$\begin{aligned}\tfrac{2\gamma-\mu}{2}\textstyle\sum_{k=s}^K \mathbb{E}\big[\|x^{k-1} - \bar{x}^k\|_2^2|\psi_{s-1}\big] &\leqslant \psi(\bar{x}^{s-1}) - \mathbb{E}\psi(\bar{x}^K) \\ &\quad + \tfrac{\gamma+\rho}{2}\textstyle\sum_{k=s}^K \mathbb{E}\big[\|x^{k-2} - \bar{x}^{k-1}\|_2^2|\pi_{s-1}\big] + \sum_{k=s}^K \zeta_{k-1}.\end{aligned}$$

It then follows that

$$\begin{aligned}\tfrac{\gamma-\mu-\rho}{2}\mathbb{E}\big[\textstyle\sum_{k=s}^{K}\|x^{k-1}-\bar{x}^{k}\|_2^2|\pi_{s-1}\big] &\leqslant \psi(\bar{x}^{s-1})-\mathbb{E}\,\psi(\bar{x}^{K})+\tfrac{\gamma+\rho}{2}\|x^{s-2}-\bar{x}^{s-1}\|_2^2+\textstyle\sum_{k=s}^{K}\zeta_{k-1}\\ &\leqslant \psi_{s-1}(\bar{x}^{s-1})-\mathbb{E}\psi(\bar{x}^{K})\\ &\quad+\tfrac{\rho}{\gamma-\mu}\left[\psi_{s-1}(x^{s-2})-\psi_{s-1}(\bar{x}^{s-1})\right]+\textstyle\sum_{k=s}^{K}\zeta_{k-1}\\ &\leqslant \psi(x^{s-2})-\mathbb{E}\psi(\bar{x}^{K})\\ &\quad+\tfrac{\rho}{\gamma-\mu}\left[\psi(x^{s-2})-\psi_{s-1}(\bar{x}^{s-1})\right]+\textstyle\sum_{k=s}^{K}\zeta_{k-1}\\ &\leqslant \tfrac{\gamma-\mu+\rho}{\gamma-\mu}\left[\psi(x^{s-2})-\psi(x^{*})\right]+\textstyle\sum_{k=s}^{K}\zeta_{k-1},\end{aligned}$$

where the third and the last inequality follow from the property

$$\psi(x^{k-1})=\psi_k(x^{k-1})\geqslant\psi_k(\bar{x}^{k})\geqslant\psi(\bar{x}^{k})\geqslant\psi(x^{*}).$$

Note that solution x^k is feasible for the k-th subproblem and hence, in view of Proposition 5.5.1, we have that $g(\bar{x}^k)\leqslant g_k(\bar{x}^k)\leqslant\eta_k<\eta$ and hence $\bar{x}^k$ is feasible solution for the main problem implying $\psi(\bar{x}^k)\geqslant\psi(x^*)$ in the above relation. Then (5.25) immediately follows.

Now we prove that (5.26) holds. Note that

$$\mathbb{E}\left[\|x^k-\bar{x}^k\|_2^2|\pi_k\right]\leqslant\tfrac{2}{\gamma-\mu}\mathbb{E}\left[\psi_k(x^k)-\psi_k(\bar{x}^k)|\pi_k\right]\leqslant\tfrac{2}{\gamma-\mu}\left[\tfrac{\rho}{2}\|x^{k-1}-\bar{x}^k\|_2^2+\zeta_k\right],$$

where first inequality follows due to strong convexity ψ_k as well as optimality of $\bar{x}^k$ and second inequality follows due to (5.9). Now summing the above relation from $k=s$ to K and taking expectation conditioned on ψ_{s-1}, we obtain

$$\begin{aligned}\mathbb{E}\left[\textstyle\sum_{k=s}^{K}\|x^k-\bar{x}^k\|_2^2|\pi_{s-1}\right]&\leqslant\tfrac{\rho}{\gamma-\mu}\mathbb{E}\left[\textstyle\sum_{k=s}^{K}\|x^{k-1}-\bar{x}^k\|_2^2|\pi_{s-1}\right]+\tfrac{2}{\gamma-\mu}\textstyle\sum_{k=s}^{K}\zeta_k\\ &\leqslant\tfrac{2\rho A_s}{(\gamma-\mu)(\gamma-\mu-\rho)}+\tfrac{2Z_s}{\gamma-\mu-\rho},\end{aligned}$$

where last inequality follows from (5.25) and definition of Z_s. Hence, we conclude the proof. □

Now we present the unified convergence of proximal point as stated in Theorem 5.3.3.

Proof of Theorem 5.3.3

Due to the KKT condition for the subproblem (5.8), we have

$$\begin{aligned} 0 &\in \partial\psi(\bar{x}^k) + \gamma\left(\bar{x}^k - x^{k-1}\right) + \bar{y}^k\left(\partial\|\bar{x}^k\|_1 - \nabla h(x^{k-1})\right) \\ 0 &= \bar{y}^k\left(\lambda\|\bar{x}^k\|_1 - h(x^{k-1}) - \langle\nabla h(x^{k-1}), \bar{x}^k - x^{k-1}\rangle - \eta_k\right) \end{aligned} \tag{5.27}$$

Using triangle inequality along with first relation in the above equation, we have $\operatorname{dist}\left(\partial_x\mathcal{L}(\bar{x}^k, \bar{y}^k), 0\right) \leqslant \gamma\|\bar{x}^k - x^{k-1}\|_2 + \bar{y}^k\|\nabla h(x^{k-1}) - \nabla h(\bar{x}^k)\|_2$. Therefore, noting the bound on $\bar{y}^k$ from Assumption 5.3.1, we have

$$\begin{aligned} \operatorname{dist}\left(\partial_x\mathcal{L}(\bar{x}^k, \bar{y}^k), 0\right)^2 &\leqslant 2\gamma^2\|\bar{x}^k - x^{k-1}\|_2^2 + 2B^2\|\nabla h(x^{k-1}) - h(\bar{x}^k)\|_2^2 \\ &\leqslant 2\left(\gamma^2 + B^2L_h^2\right)\|\bar{x}^k - x^{k-1}\|_2^2, \end{aligned}$$

where the second inequality uses Lipschitz smoothness of $h(x)$. Summing the above relation from $k = s, \ldots, K$ and the taking expectation conditioned on π_{s-1} on both sides, we obtain

$$\begin{aligned} \mathbb{E}\left[\textstyle\sum_{k=s}^K \operatorname{dist}\left(\partial_x\mathcal{L}(\bar{x}^k, \bar{y}^k), 0\right)^2 |\pi_{s-1}\right] &\leqslant 2(\gamma^2 + B^2L_h^2)\mathbb{E}\left[\textstyle\sum_{k=s}^K \|x^{k-1} - \bar{x}^k\|_2^2 |\pi_{s-1}\right] \\ &\leqslant \tfrac{4\left(\gamma^2+B^2L_h^2\right)}{\gamma-\mu-\rho}(A_s + Z_s), \end{aligned} \tag{5.28}$$

For the complementary slackness part of the KKT condition, first notice that $\eta_k = \eta_0 + \sum_{t=1}^k \delta_t = \eta_0 + \sum_{t=1}^k \frac{\eta-\eta_0}{t(t+1)} = \frac{k}{k+1}\eta + \frac{1}{k+1}\eta_0$. Therefore,

$$\textstyle\sum_{k=s}^K (\eta - \eta_k) = \sum_{k=s}^K \frac{\eta-\eta_0}{k+1} \leqslant \frac{K+1-s}{s+1}(\eta - \eta_0).$$

To prove the error of complementary slackness condition, observe that

$$\begin{aligned}\bar{y}^k \left|\lambda\|\bar{x}^k\|_1 - h(\bar{x}^k) - \eta\right| &\leqslant \bar{y}^k \left|\lambda\|\bar{x}^k\|_1 - h(x^{k-1}) - \langle \nabla h(x^{k-1}), \bar{x}^k - x^{k-1}\rangle - \eta_k\right| \\ &\quad + \bar{y}^k \left|h(x^{k-1}) + \langle \nabla h(x^{k-1}), \bar{x}^k - x^{k-1}\rangle - h(\bar{x}^k)\right| + \bar{y}^k (\eta - \eta_k) \\ &\leqslant \tfrac{BL_h}{2}\|\bar{x}^k - x^{k-1}\|_2^2 + B(\eta - \eta_k),\end{aligned}$$

where second inequality follows due to second relation in (5.27) and bound on $\bar{y}^k$ from Assumption 5.3.1. Summing the above relation from $k = s, \ldots, K$ and taking expectation conditioned on π_{s-1} on both sides, we obtain

$$\begin{aligned}\mathbb{E}\left[\textstyle\sum_{k=s}^K \bar{y}^k \left|g(\bar{x}^k) - \eta\right| | \pi_{s-1}\right] &\leqslant \textstyle\sum_{k=s}^K \mathbb{E}\left[\tfrac{BL_h}{2}\|\bar{x}^k - x^{k-1}\|_2^2 + B(\eta - \eta_k) | \pi_{s-1}\right] \\ &\leqslant \tfrac{BL_h}{2}\mathbb{E}\left[\textstyle\sum_{k=s}^K \|\bar{x}^k - x^{k-1}\|_2^2 | \psi_{s-1}\right] + B\textstyle\sum_{k=s}^K (\eta - \eta_k) \\ &\leqslant \tfrac{BL_h}{\gamma-\mu-\rho}(A_s + Z_s) + \tfrac{(K+1-s)B(\eta-\eta_0)}{s+1}. \qquad (5.29)\end{aligned}$$

Now note that $A_s = \frac{\gamma-\mu+\rho}{\gamma-\mu}[\psi(x^{s-2}) - \psi(x^*)]$ is a random variable due to randomness of x^{s-2}. Now we bound expectation of $\psi(x^{s-2})$. In view of (5.9), we have

$$\begin{aligned}\mathbb{E}[\psi_k(x^k)|\pi_k] &\leqslant \psi_k(\bar{x}^k) + \tfrac{\rho}{2}\|x^{k-1} - \bar{x}^k\|_2^2 + \zeta_k \\ &\leqslant \psi_k(x^{k-1}) - \tfrac{\gamma-\mu-\rho}{2}\|x^{k-1} - \bar{x}^k\|_1 + \zeta_k\end{aligned}$$

Since, $\gamma - \mu - \rho \geqslant 0$ and noting that $\psi_k(x^{k-1}) = \psi(x^{k-1})$, $\psi_k(x^k) \geqslant \psi(x^k)$, we have

$$\mathbb{E}[\psi(x^k)|\pi_k] \leqslant \psi(x^{k-1}) + \zeta_k.$$

Taking expectation on both sides of the above relation and then summing from $k = 1$ to $s - 2$, we get

$$\mathbb{E}[\psi(x^{s-2})] \leqslant \psi(x^0) + \textstyle\sum_{k=1}^{s-2}\zeta_k.$$

Using the above relation, we obtain

$$\mathbb{E}[A_s] \leqslant \tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2\textstyle\sum_{k=1}^{s-2}\zeta_k, \tag{5.30}$$

where $\Delta^0 = \psi(x^0) - \psi(x^*)$. Note that here we used the fact $\frac{\gamma-\mu+\rho}{\gamma-\mu} \leqslant 2$. Now taking expectation on both sides of (5.28) and using bound on $\mathbb{E}[A_s]$ in (5.30), we obtain

$$\begin{aligned}\mathbb{E}\left[\textstyle\sum_{k=s}^{K} \operatorname{dist}\left(\partial_x \mathcal{L}(\bar{x}^k, \bar{y}^k), 0\right)^2 |\pi_{s-1}\right] &\leqslant \tfrac{4(\gamma^2+B^2L_h^2)}{\gamma-\mu-\rho}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2\textstyle\sum_{k=1}^{s-2}\zeta_k + \sum_{k=s-1}^{K}\zeta_k\big) \\ &\leqslant \tfrac{4(\gamma^2+B^2L_h^2)}{\gamma-\mu-\rho}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2Z_1\big).\end{aligned}$$

Similarly, taking expectation on both sides of (5.29) and using (5.30), we obtain

$$\mathbb{E}\left[\textstyle\sum_{k=s}^{K} \bar{y}^k \left|g(\bar{x}^k) - \eta\right| |\pi_{s-1}\right] \leqslant \tfrac{BL_h}{\gamma-\mu-\rho}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2Z_1\big) + \tfrac{K+1-s}{s+1}B(\eta-\eta_0).$$

Taking expectation on both sides of (5.26) and using (5.30), we obtain

$$\begin{aligned}\mathbb{E}[\textstyle\sum_{k=s}^{K}\|x^k - \bar{x}^k\|_2^2] &\leqslant \tfrac{2\rho}{(\gamma-\mu)(\gamma-\mu-\rho)}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2\textstyle\sum_{k=1}^{s-2}\zeta_k\big) + \tfrac{2Z_s}{\gamma-\mu-\rho} \\ &\leqslant \tfrac{2\rho(\gamma-\mu+\rho)}{(\gamma-\mu)^2(\gamma-\mu-\rho)}\Delta^0 + \tfrac{4Z_1}{\gamma-\mu-\rho}.\end{aligned}$$

Finally, setting $s = \left\lfloor \frac{K+1}{2} \right\rfloor$, we have $\frac{K}{2} \leqslant s \leqslant \frac{K+1}{2}$. Therefore, we have

$$\begin{aligned}\mathbb{E}_{\hat{k}}\left[\operatorname{dist}\left(\partial_x \mathcal{L}(\bar{x}^{\hat{k}}, \bar{y}^{\hat{k}}), 0\right)^2\right] &\leqslant \tfrac{8(\gamma^2+B^2L_h^2)}{K(\gamma-\mu-\rho)}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2Z_1\big), \\ \mathbb{E}_{\hat{k}}\left[\bar{y}^{\hat{k}}\left|g(\bar{x}^{\hat{k}}) - \eta\right|\right] &\leqslant \tfrac{2BL_h}{K(\gamma-\mu-\rho)}\big(\tfrac{\gamma-\mu+\rho}{\gamma-\mu}\Delta^0 + 2Z_1\big) + \tfrac{2B(\eta-\eta_0)}{K},\end{aligned}$$

and

$$\mathbb{E}_{\hat{k}}\|x^{\hat{k}} - \bar{x}^{\hat{k}}\|_2^2 \leqslant \tfrac{4\rho(\gamma-\mu+\rho)}{K(\gamma-\mu)^2(\gamma-\mu-\rho)}\Delta^0 + \tfrac{8Z_1}{K(\gamma-\mu-\rho)}.$$

Hence, we conclude the proof.

5.5.6 Proof of Corollary 5.3.5

Since $T_k \geqslant 2\sqrt{\frac{L}{\mu}+3}$, we have that $\frac{2(L+\gamma)}{T_k^2} = \frac{2(L+3\mu)}{T_k^2} \leqslant \frac{\mu}{2} = \frac{\rho}{2}$. Moreover, we see that $\rho = \mu \leqslant \gamma - \mu = 2\mu$. Finally, since $T_k \geqslant K(M+\sigma)$ so we have $\zeta_k \leqslant \frac{4}{\mu K}$ implying that $Z_1 = \sum_{k=1}^{K} \zeta_k \leqslant \frac{4}{\mu}$. Then, applying Theorem 5.3.3, we obtain that $x^{\hat{k}}$ is an $(\varepsilon_1, \varepsilon_2)$-KKT solution of the problem (5.5).

5.5.7 Convergence for the (stochastic) convex case

We have the following Corollary of Theorem 5.3.3 for the case in which objective ψ is convex, i.e. $\mu = 0$.

Corollary 5.5.10 *Let ψ be convex function such that it satisfies* (5.6) *with $\mu = 0$. Set $\gamma = \beta L$ where $\beta \in [0, 1)$ be a small constant and run AC-SA for $T_k = \max\{2\sqrt{\frac{2(1+\beta)}{\beta}}, K(M+\sigma)\}$ iterations where K is total number of iterations of Algorithm 4. Then, we obtain that $x^{\hat{k}}$ is an $(\varepsilon_1, \varepsilon_2)$-KKT point of the problem* (5.5) *where*

$$\varepsilon_1 = \left(\tfrac{3\Delta^0}{2K} + \tfrac{16(M+\sigma)}{\beta K L}\right) \max\{\tfrac{16(\beta^2 L^2 + B^2 L_h^2)}{\beta L}, \tfrac{4BL_h}{\beta L}\} + \tfrac{2B(\eta - \eta_0)}{K},$$

$$\varepsilon_2 = \tfrac{3\Delta^0}{2\beta L K} + \tfrac{128(M+\sigma)}{\beta L^2 K}.$$

Proof. Since $T_k \geqslant 2\sqrt{\frac{2(1+\beta)}{\beta}}$, we have $\frac{2(L+\gamma)}{T_k^2} = \frac{2(1+\beta)L}{T_k^2} \leqslant \frac{\beta L}{4} = \frac{\rho}{2}$. Moreover, note that $\rho = \frac{\beta L}{2} \leqslant \gamma = \beta L$. Finally, since $T_k \geqslant K(M+\sigma)$ so we have $\zeta_k = \frac{8(M^2+\sigma^2)}{\gamma T_k} \leqslant \frac{8(M+\sigma)}{\beta L K}$. Hence, $Z_1 = \sum_{k=1}^{K} \zeta_k \leqslant \frac{8(M+\sigma)}{\beta L}$. Then, applying Theorem 5.3.3, we obtain that $x^{\hat{k}}$ is an $(\varepsilon_1, \varepsilon_2)$-KKT solution of problem (5.5). □

Finite-sum problem A special case of objective takes the finite-sum form $f(x) = \frac{1}{n}\sum_{i=1}^{n} \widetilde{f}_i(x)$ thereby leading to the following subproblem

$$\min_x \widetilde{\psi}(x) = \tfrac{1}{n}\textstyle\sum_{i=1}^{n} \widetilde{f}_i(x) + \widetilde{\omega}(x)$$

It is known that finite-sum problem can be efficiently solved by using variance reduction or randomized incremental gradient method [112, 59]. The complexity of LCPP on finite-sum problem can be further improved if we apply variance reduction technique for solving the subproblem. We comment on the complexity result in brief. In the finite-sum setting, the Nesterov's accelerated gradient-based LCPP requires $T_k = \widetilde{\mathcal{O}}(n\sqrt{\frac{L+2\mu}{\mu}})$ and $T_k = \widetilde{\mathcal{O}}(n\beta^{-1/2})$ number of stochastic gradient computations to solve each LCPP subproblem. Even though this number is a constant in terms of dependence on K, number of terms (n) in the finite sum can be large. In comparison to these standard methods, the complexity of SVRG (stochastic variance reduced gradient) based LCPP method can be improved to $T_k = \widetilde{\mathcal{O}}(n + \frac{L+\mu}{\mu})$ for the case when ψ is nonconvex satisfying (5.6) with $\mu > 0$, and to $T_k = \widetilde{\mathcal{O}}(n + \beta^{-1})$ for convex problem where $\mu = 0$. This will be verified in the numerical experiments section.

5.5.8 Proof for the projection algorithm for problem (5.11)

Here, we describe an efficient algorithm for solving the (5.11). Specifically, we formulate the update as the following problem

$$\min_{x\in\mathbb{R}^d} \tfrac{1}{2}\|x - v\|_2^2 \text{ s.t. } \|x\|_1 + \langle u, x\rangle \leqslant \tau. \tag{5.31}$$

Since the objective is strongly convex, problem (5.31) has a unique global optimal solution. Moreover, the problem is strictly feasible because of the strict feasibility guarantee (5.5.1) in the context of problem (5.8). Therefore, KKT condition guarantees that there exists $y \geqslant 0$ such that

$$0 \in x - v + yu + y\partial\|x\|_1, \tag{5.32}$$

$$0 = y\left(\langle u, x\rangle + \|x\|_1 - \tau\right). \tag{5.33}$$

The algorithm proceeds as follows. First, we check whether v is feasible, if it is the case, then $x = v$ is the optimal solution. Otherwise, the constraint in (5.31) is active. Next, we explore the optimality condition (5.32). Given the optimal Lagrangian multiplier $y \geqslant 0$, for the i-th coordinate

of the optimal x, one of the following three situations will occur:

1. $x_i > 0$ and $x_i = v_i - (u_i + 1)y$.
2. $x_i < 0$ and $x_i = v_i - (u_i - 1)y$.
3. $x_i = 0$ and $(u_i - 1)y \leqslant v_i \leqslant (u_i + 1)y$.

For simplicity, let us denote $[a]_+ = \max\{a, 0\}$ and $[a, b]_+ = \max\{a, b, 0\}$. Based on the discussion above, we can express x as a piecewise linear function of y.

$$x_i(y) = [v_i - (u_i + 1)y]_+ - [(u_i - 1)y - v_i]_+ .$$

Let us denote $\ell(y) = \langle u, x(y)\rangle + \|x(y)\|_1$. We can deduce that

$$\begin{aligned}
\ell(y) &= \textstyle\sum_{i=1}^d u_i x_i(y) + \sum_{i=1}^d \max\{x_i(y), -x_i(y)\} \\
&= \textstyle\sum_{i=1}^d u_i [v_i - (u_i + 1)y]_+ - \sum_{i=1}^d u_i [(u_i - 1)y - v_i]_+ \\
&\quad + 2\textstyle\sum_{i=1}^d [v_i - (u_i + 1)y, (u_i - 1)y - v_i]_+ \\
&\quad - \textstyle\sum_{i=1}^d [v_i - (u_i + 1)y]_+ - \sum_{i=1}^d [(u_i - 1)y - v_i]_+ \\
&= \textstyle\sum_{i=1}^d (u_i - 1) [v_i - (u_i + 1)y]_+ \\
&\quad - \textstyle\sum_{i=1}^d (u_i + 1) [(u_i - 1)y - v_i]_+ \\
&\quad + 2\textstyle\sum_{i=1}^d [v_i - (u_i + 1)y, (u_i - 1)y - v_i]_+
\end{aligned}$$

Above, the second equality uses the identity: $\max\{p - q, q - p\} = 2\max\{p, q\} - p - q$ for any $p, q \in \mathbb{R}$. It can be readily seen that $\ell(y)$ is a piecewise linear function with at most $3d$ breaking points. We can sort these points in $\mathcal{O}(d \log d)$ and then apply a line-search to find the root of $\ell(\cdot) = \tau$ in $\mathcal{O}(d)$ time.

5.5.9 Supermartingale convergence theorem

In below, we state a version of supermartingale convergence theorem developed by [94].

Theorem 5.5.11 *Let (Ω, F, P) be a probability space and $\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq ... \subseteq \mathcal{F}_k \subseteq$ be some sub-σ-algebra of F. Let b_k, c_k be nonnegative $\mathcal{F}_k$-measurable random variables such that*

$$\mathbb{E}\left[b_{k+1} \mid \mathcal{F}_k\right] \leqslant b_k + \xi_k - c_k,$$

where $\{\xi_k\}_{0 \leqslant k < \infty}$ is a non-negative and summable: $\sum_{k=0}^{\infty} \xi_k < +\infty$. Then we have

$$\lim_{k \to \infty} b_k \text{ exists, and } \sum_{k=1}^{\infty} c_k < +\infty, \quad a.s.$$

CHAPTER 6

FASTER WIDTH-DEPENDENT ALGORITHM FOR MIXED PACKING AND COVERING LPS

In chapter 3, we saw a primal-dual type algorithm for solving function constrained optimization problem. In that problem, we assumed that the primal feasible set X is a simple set whose radius is not too big. However, for certain important class of linear programs (LPs), we need to set X to be an ℓ_∞-ball. Such LPs arise quite naturally in combinatorial optimization and hence require special attention. Note that the radius is of an ℓ_∞-ball is at least $\Omega(\sqrt{n})$ where n is the dimension of LP which can be quite large for many practical applications. In this chapter, we focus on this well-known ℓ_∞ barrier and propose a new algorithm that can overcome it.

6.1 Mixed Packing and Covering LPs

Mixed packing and covering linear programs (LPs) are a natural class of LPs where coefficients, variables, and constraints are non-negative. They model a wide range of important problems in combinatorial optimization and operations research. In general, they model any problem which contains a limited set of available resources (packing constraints) and a set of demands to fulfill (covering constraints).

Two special cases of the problem have been widely studied in literature: pure *packing*, formulated as $\max_x\{b^T x \mid Px \leqslant p\}$; and pure *covering*, formulated as $\min_x\{b^T x \mid Cx \geqslant c\}$ where P, p, C, c, b are all non-negative. These are known to model fundamental problems such as maximum bipartite graph matching, minimum set cover, etc. [69]. Algorithms to solve packing and covering LPs have also been applied to great effect in designing flow control systems [8], scheduling problems [89], zero-sum matrix games [80] and in mechanism design [124]. In this paper, we study the mixed packing and covering (MPC) problem, formulated as checking the feasibility of the set: $\{x \mid Px \leqslant p, Cx \geqslant c\}$, where P, C, p, c are non-negative. We say that x is an ε-

approximate solution to MPC if it belongs to the relaxed set $\{x \mid Px \leqslant (1+\varepsilon)p, Cx \geqslant (1-\varepsilon)c\}$. MPC is a generalization of pure packing and pure covering, hence it is applicable to a wider range of problems such as multi-commodity flow on graphs [115, 100], non-negative linear systems and X-ray tomography [115].

General LP solving techniques such as the interior point method can approximate solutions to MPC in as few as $O(\log(1/\varepsilon))$ iterations - however, they incur a large per-iteration cost. In contrast, iterative approximation algorithms based on first-order optimization methods require poly$(1/\varepsilon)$ iterations, but the iterations are fast and in most cases are conducive to efficient parallelization. This property is of utmost importance in the context of ever-growing datasets and the availability of powerful parallel computers, resulting in much faster algorithms in relatively low-precision regimes.

6.1.1 Previous work

In literature, algorithms for the MPC problem can be grouped into two broad categories: *width-dependent* and *width-independent*. Here, *width* is an intrinsic property of a linear program which typically depends on the dimensions and the largest entry of the constraint matrix, and is an indication of the range of values any constraint can take. In the context of this paper and the MPC problem, we define w_P and w_C as the maximum number of non-zeros in any constraint in P and C respectively. We define the width of the LP as $w := \max(w_P, w_C)$.

One of the first approaches used to solve LPs was Langrangian-relaxation: replacing hard constraints with loss functions which enforce the same constraints indirectly. Using this approach, Plotkin, Schmoys and Tardos [89], and Grigoriadis and Khachiyan [44] obtained width-dependent polynomial-time approximation algorithms for MPC. Luby and Nisan [69] gave the first width-dependent parallelizable algorithm for pure packing and pure covering, which ran in $\widetilde{O}(\varepsilon^{-4})$ parallel time, and $\widetilde{O}(N\varepsilon^{-4})$ total work. Here, *parallel time* (sometimes termed as *depth*) refers to the longest chain of dependent operations, and *work* refers to the total number of operations in the algorithm.

Young [115] extended this technique to give the first width-independent parallel algorithm for MPC in $\widetilde{O}(\varepsilon^{-4})$ parallel time, and $\widetilde{O}(md\varepsilon^{-2})$ total work[1]. Young [116] later improved his algorithm to run using total work $O(N\varepsilon^{-2})$. Mahoney *et al.* [71] later gave an algorithm with a faster parallel run-time of $\widetilde{O}(\varepsilon^{-3})$.

The other most prominent approach in literature towards solving an LP is by converting it into a smooth function [80], and then applying general first-order optimization techniques [80, 82]. Although the dependence on ε from using first-order techniques is much improved, it usually comes at the cost of sub-optimal dependence on the input size and width. For the MPC problem, Nesterov's accelerated method [82], as well as Bienstock and Iyengar's adaptation [14] of Nesterov's smoothing [80], give rise to algorithms with runtime linearly depending on ε^{-1}, but with far from optimal dependence on input size and width. For pure packing and pure covering problems, however, Allen-Zhu and Orrechia [2] were the first to incorporate Nesterov-like acceleration while still being able to obtain near-linear width-independent runtimes, giving a $\widetilde{O}(N\varepsilon^{-1})$ time algorithm for the packing problem. For the covering problem, they gave a $\widetilde{O}(N\varepsilon^{-1.5})$ time algorithm, which was then improved to $\widetilde{O}(N\varepsilon^{-1})$ by [107]. Importantly, however, the above algorithms do not generalize to MPC.

6.1.2 Our contributions

We give the best parallel width-dependent algorithm for MPC, while only incurring a linear dependence on ε^{-1} in the parallel runtime and total work. Additionally, the total work has near-linear dependence on the input-size. Formally, we state our main theorem as follows.

Theorem 6.1.1 *There exists a parallel ε-approximation algorithm for the mixed packing covering problem, which runs in $\widetilde{O}(w \cdot \varepsilon^{-1})$ parallel time, while performing $\widetilde{O}(w \cdot N \cdot \varepsilon^{-1})$ total work, where N is the total number of non-zeros in the constraint matrices, and w is the width of the given LP.*

Table 6.1 compares the running time of our algorithm to previous works solving this problem.

[1] d here is the maximum number of constraints that any variable appears in.

Table 6.1: Comparison of runtimes of ε-approximation algorithms for the mixed packing covering problem.

	Parallel Runtime	**Total Work**	**Comments**
Young [115]	$\widetilde{O}(\varepsilon^{-4})$	$\widetilde{O}(md\varepsilon^{-2})$	d is column-width
Bienstock and Iyengar [14]		$\widetilde{O}(n^{2.5}w_P^{1.5}w\varepsilon^{-1})$	width-dependent
Nesterov [82]	$\widetilde{O}(w\sqrt{n}\varepsilon^{-1})$	$\widetilde{O}(w \cdot N\sqrt{n}\varepsilon^{-1})$	width-dependent
Young [116]	$\widetilde{O}(\varepsilon^{-4})$	$\widetilde{O}(N\varepsilon^{-2})$	
Mahoney *et al.* [71]	$\widetilde{O}(\varepsilon^{-3})$	$\widetilde{O}(N\varepsilon^{-3})$	
This paper	$\widetilde{O}(w\varepsilon^{-1})$	$\widetilde{O}(wN\varepsilon^{-1})$	width-dependent

Sacrificing width independence for faster convergence with respect to precision proves to be a valuable trade-off for several combinatorial optimization problems which naturally have a low width. Prominent examples of such problems which are not pure packing or covering problems include *multicommodity flow* and *densest subgraph*, where the width is bounded by the degree of a vertex. In a large number of real-world graphs, the maximum vertex degree is usually small, hence our algorithm proves to be much faster when we want high-precision solutions. We explicitly show that this result directly gives the fastest algorithm for the densest subgraph problem on low-degree graphs in Section 6.5.12.

6.2 Notation and Definitions

For any integer q, we represent using $\|\cdot\|_q$ the q-norm of any vector. We represent the infinity-norm as $\|\cdot\|_\infty$. We denote the infinity-norm ball (sometimes called the ℓ_∞ ball) as the set $\mathcal{B}_\infty^n(r) := \{x \in \mathbb{R}^n : \|x\|_\infty \leqslant r\}$. The nonnegative part of this ball is denoted as $\mathcal{B}_{+,\infty}^n(r) = \{x \in \mathbb{R}^n : x \geqslant \mathbf{0}_n, \|x\|_\infty \leqslant r\}$. For radius $r = 1$, we drop the radius specification and use the short notation $\mathcal{B}_\infty^n$ and $\mathcal{B}_{+,\infty}^n$. We denote the extended simplex of dimension k as $\Delta_k^+ := \{x \in \mathbb{R}^k : \sum_{i=1}^k x_i \leqslant 1\}$. For any $y \geqslant \mathbf{0}_k$, $\mathrm{proj}_{\Delta_k^+}(y) = y/\|y\|_1$ if $\|y\|_1 \geqslant 1$. Further, for any set K, we represent its interior, relative interior and closure as $\mathrm{int}(K), \mathrm{relint}(K)$ and $\mathrm{cl}(K)$, respectively. The function $\exp$ is applied to a vector element wise. The division of two vectors of same dimension is also performed

element wise.

For any matrix A, we use $\mathrm{nnz}(A)$ to denote the number of nonzero entries in it. We use $A_{i,:}$ and $A_{:,j}$ to refer to the ith row and jth column of A respectively. We use notation A_{ij} (or $A_{i,j}$ alternatively) to denote an element in the i-th row and j-th column of matrix A. $\|A\|_\infty$ denotes the operator norm $\|A\|_{\infty\to\infty} := \sup_{x\neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty}$. For a symmetric matrix A and an antisymmetric matrix B, we define an operator $\succeq_{\mathrm{i}}$ as $A \succeq_{\mathrm{i}} B \Leftrightarrow \begin{bmatrix} A & -B \\ B & A \end{bmatrix}$ is positive semi-definite.

We formally define an ε-approximate solution to the mixed packing-covering (MPC) problem as follows.

Definition 6.2.1 *We say that x is an ε-approximate solution of the mixed packing-covering problem if x satisfies $x \in \mathcal{B}^n_{+,\infty}$, $Px \leqslant (1+\varepsilon)\mathbf{1}_p$ and $Cx \geqslant (1-\varepsilon)\mathbf{1}_c$.*

Here, $\mathbf{1}_k$ denotes a vectors of 1's of dimension k for any integer k.

The saddle point problem on two sets $x \in X$ and $y \in Y$ can be defined as follows:

$$\min_{x\in X} \max_{y\in Y} \mathcal{L}(x,y), \tag{6.1}$$

where $\mathcal{L}(x,y)$ is some bilinear form between x and y. For this problem, we define the *primal-dual gap function* as $\sup_{(\bar{x},\bar{y})\in X\times Y} \mathcal{L}(x,\bar{y}) - \mathcal{L}(\bar{x},y)$. This gap function can be used as measure of accuracy of the above saddle point solution.

Definition 6.2.2 *We say that $(x,y) \in X\times Y$ is an ε-optimal solution for* (6.1) *if* $\sup_{(\bar{x},\bar{y})\in X\times Y} \mathcal{L}(x,\bar{y}) - \mathcal{L}(\bar{x},y) \leqslant \varepsilon$.

6.3 Technical overview

The mixed packing-covering (MPC) problem is formally defined as follows.

Given two nonnegative matrices $P \in \mathbb{R}^{p\times n},\ C \in \mathbb{R}^{c\times n}$, find an $x \in \mathbb{R}^n, x \geqslant \mathbf{0}, \|x\|_\infty \leqslant 1$ such that $Px \leqslant \mathbf{1}_p$ and $Cx \geqslant \mathbf{1}_c$ if it exists, otherwise report infeasibility.

Note that the vector of 1's on the right hand side of the packing and covering constraints can be obtained by simply scaling each constraint appropriately. We also assume that each entry in the matrices P and C is at most 1. This assumption, and subsequently the ℓ_∞ constraints on x also cause no loss of generality[2].

We reformulate MPC as a saddle point problem, as defined in Section 6.2;

$$\lambda^* := \min_{x \in \mathcal{B}^n_{+,\infty}} \max_{y \in \Delta^+_c,\ z \in \Delta^+_p} L(x, y, z), \tag{6.2}$$

where $L(x, y, z) := [y^T\ z^T] \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix}$. The relation between the two formulations is shown in Section 6.4. For the rest of the paper, we focus on the saddle point formulation (6.2).

$\eta(x) := \max_{y \in \Delta^+_c, z \in \Delta^+_p} L(x, y, z)$ is a piecewise linear convex function. Assuming oracle access to this "inner" maximization problem, the "outer" problem of minimizing $\eta(x)$ can be performed using first order methods like mirror descent, which are suitable when the underlying problem space is the unit ℓ_∞ ball. One drawback of this class of methods is that their rate of convergence, which is standard for non-accelerated first order methods on non-differentiable objectives, is $O(\frac{1}{\varepsilon^2})$ to obtain an ε-approximate minimizer x of η which satisfies $\eta(x) \leqslant \eta^* + \varepsilon$, where η^* is the optimal value. This means that the algorithm needs to access the inner maximization oracle $O(\frac{1}{\varepsilon^2})$ times, which can become prohibitively large in the high precision regime.

Note that even though η is a piecewise linear non-differentiable function, it is not a black box function, but a maximization linear functions in x. This structure can be exploited using Nesterov's smoothing technique [80]. In particular, $\eta(x)$ can be approximated by choosing a strongly convex[3] function $\phi : \Delta^+_p \times \Delta^+_c \to \mathbb{R}$ and considering

$$\widetilde{\eta}(x) = \max_{y \in \Delta^+_c, z \in \Delta^+_p} L(x, y, z) - \phi(y, z).$$

[2]This transformation can be achieved by adapting techniques from [107] while increasing dimension of the problem up to a logarithmic factor. Details of this fact are in Appendix 6.5.11 in the full version of this paper. For the purpose of the main text, we work with this assumption.

This strongly convex regularization yields that $\widetilde{\eta}$ is a Lipschitz-smooth[3] convex function. If L is the constant of Lipschitz smoothness of $\widetilde{\eta}$ then application of any of the accelerated gradient methods in literature will converge in $O(\sqrt{\frac{L}{\varepsilon}})$ iterations. Moreover, it can also be shown that in order to construct a smooth ε-approximation $\widetilde{\eta}$ of η, the Lipschitz smoothness constant L can be chosen to be of the order $O(\frac{1}{\varepsilon})$, which in turn implies an overall convergence rate of $O(\frac{1}{\varepsilon})$. In particular, Nesterov's smoothing achieves an oracle complexity of $O((\|P\|_\infty^+\|C\|_\infty^)D_x \max\{D_y, D_z\}\varepsilon^{-1})$, where where D_x, D_y and D_z denote the sizes of the ranges of their respective regularizers which are strongly convex functions. D_y and D_z can be made of the order of $\log p$ and $\log c$, respectively. However, D_x can be problematic since x belongs to an ℓ_∞ ball. More on this will soon follow.

Nesterov's dual extrapolation algorithm [81] gives a very similar complexity but is a different algorithm in that it directly addresses the saddle point formulation (6.2) rather than viewing the problem as optimizing a non-smooth function η. The final convergence for the dual extrapolation algorithm is given in terms of the *primal-dual gap* function of the saddle point problem (6.2). This algorithms views the saddle point problem as solving variational inequality for an appropriate monotone operator in joint domain (x, y, z). Moreover, as opposed to smoothing techniques which only regularize the dual, this algorithm regularizes both primal and dual parts *(joint regularization)*, hence is a different scheme altogether.

Note that for both schemes mentioned above, the maximization oracle itself has an analytical expression which involves matrix-vector multiplication. Hence each call to the oracle incurs a sequential run-time of $\text{nnz}(P) + \text{nnz}(C)$. Then, overall complexity for both schemes is of order $O((\text{nnz}(P) + \text{nnz}(C))(\|P\|_\infty^+\|C\|_\infty^)D_x \max\{D_y, D_z\}\varepsilon^{-1})$.

6.3.1 The ℓ_∞ barrier

Note that the both methods, i.e., Nesterov's smoothing and dual extrapolation, involves a D_x term, which denotes the range of a convex function over the domain of x. The following lemma states a

[3]Definitions of Lipschitz-smoothness and strong convexity can be found in many texts in nonlinear programming and machine learning. e.g. [18]. Intuitively, f is Lipschitz-smooth if the rate of change of ∇f can be bounded by a quantity known as the "constant of Lipschitz smoothness".

lower bound for this range in case of ℓ_∞ balls.

Lemma 6.3.1 *Any strongly convex function has a range of at least $\Omega(\sqrt{n})$ on any ℓ_∞ ball.*

Since $D_x = \Omega(\sqrt{n})$ for each member function of this wide class, there is no hope of eliminating this $\sqrt{n}$ factor using techniques involving explicit use of strong convexity.

So, the goal now is to find a joint regularization function with a small range over ℓ_∞ balls, but still act as good enough regularizers to enable accelerated convergence of the descent algorithm. In pursuit of breaking this ℓ_∞ barrier, we draw inspiration from the notion of *area convexity* introduced by Sherman [100]. Area convexity is a weaker notion than strong convexity, however, it is still strong enough to ensure that accelerated first order methods still go through when using area convex regularizers. Since this is a weaker notion than strong convexity, we can construct area convex functions which have range of $O(n^{o(1)})$ on ℓ_∞ ball.

First, we define area convexity, and then go on to mention its relevance to the saddle point problem (6.2).

Area convexity is a notion defined in context of a matrix $A \in \mathbb{R}^{a\times b}$ and a convex set $K \subseteq \mathbb{R}^{a+b}$. Let $M_A := \begin{bmatrix} \mathbf{0}_{b\times b} & -A^T \\ A & \mathbf{0}_{a\times a} \end{bmatrix}$.

Definition 6.3.1 ([100]) *A function ϕ is* area convex *with respect to a matrix A on a convex set K iff for any $t, u, v \in K$, ϕ satisfies $\phi\big(\frac{t+u+v}{3}\big) \leqslant \frac{1}{3}\big(\phi(t)+\phi(u)+\phi(v)\big) - \frac{1}{3\sqrt{3}}(v-u)^T M_A (u-t)$.*

To understand the definition above, let us first look at the notion of strong convexity. ϕ is said to be strongly convex if for any two points t, u, $\frac{1}{2}(\phi(t) + \phi(u))$ exceeds $\phi(\frac{1}{2}(t+u))$ by an amount proportional to $\|t-u\|_2^2$. Definition 6.3.1 generalizes this notion in context of matrix A for any three points x, y, z. ϕ is area-convex on set K if for any three points $t, u, v \in K$, we have $\frac{1}{3}(\phi(t) + \phi(u) + \phi(v))$ exceeds $\phi(\frac{1}{3}(t+u+v))$ by an amount proportional to the area of the triangle defined by the convex hull of t, u, v.

Consider the case that points t, u, v are collinear. For this case, the area term (i.e., the term involving M_A) in Definition 6.3.1 is 0 since matrix M_A is antisymmetric. In this sense, area

convexity is even weaker than strict convexity. Moreover, the notion of area is parameterized by matrix A. To see a specific example of this notion of area, consider $A = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ and $t, u, v \in \mathbb{R}^2$. Then, for all possible permutations of t, u, v, the area term takes a value equal to $\pm(t_1(u_2 - v_2) + u_1(v_2 - t_2) + v_1(t_2 - u_2))$. Since the condition holds irrespective of the permutation so we must have that $\phi(\frac{t+u+v}{3}) \leqslant \frac{1}{3}\big(\phi(t)+\phi(u)+\phi(v)\big) - \frac{1}{3\sqrt{3}}|t_1(u_2-v_2)+u_1(v_2-t_2)+v_1(t_2-u_2)|$. But note that area of triangle formed by points t, u, v is equal to $\frac{1}{2}|t_1(u_2 - v_2) + u_1(v_2 - t_2) + v_1(t_2 - u_2)|$. Hence the area term is just a high dimensional matrix based generalization of the area of a triangle.

Coming back to the saddle point problem (6.2), we need to pick a suitable area convex function ϕ on the set $\mathcal{B}^n_{+,\infty} \times \Delta_p^+ \times \Delta_c^+$. Since ϕ is defined on the joint space, it has the property of joint regularization vis a vis (6.2). However, we need an additional parameter: a suitable matrix M_A. The choice of this matrix is related to the bilinear form of the *primal-dual gap function* of (6.2). We delve into the technical details of this in Section 6.4, however, we state that the matrix is composed of P, C and some additional constants. The algorithm we state exactly follows Nesterov's dual extrapolation method described earlier. One notable difference is that in [81], they consider joint regularization by a strongly convex function which does not depend on the problem matrices P, C but only on the constraint set $\mathcal{B}^n_{+,\infty} \times \Delta_p^+ \times \Delta_c^+$. Our area convex regularizer, on the other hand, is tailor made for the particular problem matrices P, C as well as the constraint set.

6.4 Area Convexity for Mixed Packing Covering LPs

In this section, we present our technical results and algorithm for the MPC problem, with the end goal of proving Theorem 6.1.1. First, we relate an $(1 + \varepsilon)$-approximate solution to the saddle point problem to an ε-approximate solution to MPC. Next, we present some theoretical background towards the goal of choosing and analyzing an appropriate area-convex regularizer in the context of the saddle point formulation, where the key requirement of the area convex function is to obtain a provable and efficient convergence result. Finally, we explicitly show an area convex function which is generated using a simple "gadget" function. We show that this area convex function

satisfies all key requirements and hence achieves the desired accelerated rate of convergence. This section closely follows [100], in which the author chooses an area convex function specific to the undirected multicommodity flow problem. Due to space constraints, we relegate almost all proofs to Appendix 6.5 (in the full version) and simply include pointers to proofs in [100] when it is directly applicable.

6.4.1 Saddle Point Formulation for MPC

Consider the saddle point formulation in (6.2) for MPC. Given a feasible primal-dual feasible solution pair (x, y, z) and $(\bar{x}, \bar{y}, \bar{z})$ for (6.2), we denote $w = (x, u, y, z)$ and $\bar{w} = (\bar{x}, \bar{u}, \bar{y}, \bar{z})$ where $u, \bar{u} \in \mathbb{R}$. Then, we define a function $Q : \mathbb{R}^{n+1+p+c} \times \mathbb{R}^{n+1+p+c} \to \mathbb{R}$ as

$$Q(w, \bar{w}) := [\bar{y}^T \ \bar{z}^T] \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} - [y^T \ z^T] \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix} \begin{bmatrix} \bar{x} \\ \bar{u} \end{bmatrix}.$$

Note that if $u = \bar{u} = 1$, then

$$\sup_{\bar{w} \in \mathcal{W}} Q(w, \bar{w}) = \sup_{\bar{x} \in \mathcal{B}^n_{+,\infty}, \bar{y} \in \Delta^+_p, \bar{z} \in \Delta^+_c} L(x, \bar{y}, \bar{z}) - L(\bar{x}, y, z)$$

is precisely the primal-dual gap function defined in Section 6.2. Notice that if (x^*, y^*, z^*) is a saddle point of (6.2), then we have

$$L(x^*, y, z) \leqslant L(x^*, y^*, z^*) \leqslant L(x, y^*, z^*)$$

for all $x \in \mathcal{B}^n_{+,\infty}, y \in \Delta^+_p, z \in \Delta^+_c$. From above equation, it is clear that $Q(w, w^*) \geqslant 0$ for all $w \in \mathcal{W}$ where $\mathcal{W} := \mathcal{B}^n_{+,\infty} \times \{1\} \times \Delta^+_p \times \Delta^+_c$ and $w^* = (x^*, 1, y^*, z^*) \in \mathcal{W}$. Moreover, $Q(w^*, w^*) = 0$. This motivates the following accuracy measure of the candidate approximate solution w.

Definition 6.4.1 *We say that $w \in \mathcal{W}$ is an ε-optimal solution of* (6.2) *iff*

$$\sup_{\bar{w} \in \mathcal{W}} Q(w, \bar{w}) \leqslant \varepsilon.$$

Remark 6.4.1 *Recall the definition of M_A for a matrix A in Section 6.3. We can rewrite $Q(w, \bar{w}) = \bar{w}^T J w$ where $J = M_H$ and*

$$H = \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix} \quad \Rightarrow \quad J := \begin{bmatrix} \mathbf{0}_{n\times n} & \mathbf{0}_{n\times 1} & -P^T & C^T \\ \mathbf{0}_{1\times n} & 0 & \mathbf{1}_p^T & -\mathbf{1}_c^T \\ P & -\mathbf{1}_p & \mathbf{0}_{p\times p} & \mathbf{0}_{p\times c} \\ -C & \mathbf{1}_c & \mathbf{0}_{c\times p} & \mathbf{0}_{c\times c} \end{bmatrix}.$$

Thus, the gap function in Definition 6.4.1 can be written in the bilinear form $\sup_{\bar{w}\in\mathcal{W}} \bar{w}^T J w$.

Lemma 6.4.2 relates the ε-optimal solution of (6.2) to the ε-approximate solution to MPC.

Lemma 6.4.2 *Let (x, y, z) satisfy $\sup_{(\bar{x},\bar{y},\bar{z})\in\mathcal{B}^n_{+,\infty}\times\Delta_p^+\times\Delta_c^+} L(x, \bar{y}, \bar{z}) - L(\bar{x}, y, z) \leqslant \varepsilon$. Then either*

1. x is an ε-approximate solution of MPC, or

2. y, z satisfy $y^T(P\bar{x} - \mathbf{1}_p) + z^T(-C\bar{x} + \mathbf{1}_c) > 0$ for all $\bar{x} \in \mathcal{B}^n_{+,\infty}$.

This lemma states that in order to find an ε-approximate solution of MPC, it suffices to find ε-optimal solution of (6.2). Henceforth, we will focus on ε-optimality of the saddle point formulation (6.2).

6.4.2 Area Convexity with Saddle Point Framework

Here we state some useful lemmas which help in determining whether a differentiable function is area convex. We start with the following remark which follows from the definition of area convexity (Definition 6.3.1).

Remark 6.4.3 *If ϕ is area convex with respect to A on a convex set K, and $\bar{K} \subseteq K$ is a convex set, then ϕ is area convex with respect to A on $\bar{K}$.*

The following two lemmas from [100] provide the key characterization of area convexity.

Lemma 6.4.4 *Let $A \in \mathbb{R}^{2\times 2}$ symmetric matrix.* $A \succeq_{\mathrm{i}} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \Leftrightarrow A \succeq 0$ *and* $\det(A) \geqslant 1$.

Lemma 6.4.5 *Let ϕ be twice differentiable on the interior of convex set K, i.e.,* $\mathrm{int}(K)$.

1. *If ϕ is area convex with respect to A on* $\mathrm{int}(K)$*, then $d^2\phi(x) \succeq_{\mathrm{i}} M_A$ for all $x \in \mathrm{int}(K)$.*
2. *If $d^2\phi(x) \succeq_{\mathrm{i}} M_A$ for all $x \in \mathrm{int}(K)$, then ϕ is area convex with respect to $\frac{1}{3}A$ on* $\mathrm{int}(K)$*. Moreover, if ϕ is continuous on* $\mathrm{cl}(K)$*, then ϕ is area convex with respect to $\frac{1}{3}A$ on* $\mathrm{cl}(K)$.

In order to handle the operator $\succeq_{\mathrm{i}}$ (recall from Section 6.2), we state some basic but important properties of this operator, which will come in handy in later proofs.

Remark 6.4.6 *For symmetric matrices A and C and antisymmetric matrices B and D,*

1. *If $A \succeq_{\mathrm{i}} B$ then $A \succeq_{\mathrm{i}} (-B)$.*
2. *If $A \succeq_{\mathrm{i}} B$ and $\lambda \geqslant 0$ then $\lambda A \succeq_{\mathrm{i}} \lambda B$.*
3. *If $A \succeq_{\mathrm{i}} B$ and $C \succeq_{\mathrm{i}} D$ then $A + C \succeq_{\mathrm{i}} (B + D)$.*

Having laid a basic foundation for area convexity, we now focus on its relevance to solving the saddle point problem (6.2). Considering Remark 6.4.1, we can write the gap function criterion of optimality in terms of bilinear form of the matrix J. Suppose we have a function ϕ which is area convex with respect to H on set $\mathcal{W}$. Then, consider the following *jointly-regularized* version of the bilinear form:

$$\widetilde{\eta}(w) := \sup_{\bar{w}\in\mathcal{W}} \bar{w}^T J w - \phi(\bar{w}). \tag{6.3}$$

Similar to Nesterov's dual extrapolation, one can attain $O(1/\varepsilon)$ convergence of accelerated gradient descent for function $\widetilde{\eta}(w)$ in (6.3) over variable w. In order to obtain gradients of $\widetilde{\eta}(w)$, we need access to $\mathrm{argmax}_{\bar{w}\in\mathcal{W}} \bar{w}^T J w - \phi(\bar{w})$. However, it may not be possible to find an exact maximizer in all cases. Again, one can get around this difficulty by instead using an approximate optimization oracle of the problem in (6.3).

Definition 6.4.2 *A δ-optimal solution oracle (OSO) for $\phi : \mathcal{W} \to \mathbb{R}$ takes input a and outputs $w \in \mathcal{W}$ such that*

$$a^T w - \phi(w) \geqslant \sup_{\bar{w} \in \mathcal{W}} a^T \bar{w} - \phi(\bar{w}) - \delta.$$

Given Φ as a δ-OSO for a function ϕ, consider the following algorithm (Algorithm 5):

Algorithm 5 Area Convex Mixed Packing Covering (AC-MPC)

Initialize $w_0 = (\mathbf{0}_n, 1, \mathbf{0}_{p+c})$

for $t = 0, \ldots, T$ **do**

$w_{t+1} \leftarrow w_t + \Phi(Jw_t + 2J\Phi(Jw_t))$

end for

For Algorithm 5, [100] shows the following:

Lemma 6.4.7 *Let $\phi : \mathcal{W} \to [-\rho, 0]$. Suppose ϕ is area convex with respect to $2\sqrt{3}H$ on $\mathcal{W}$. Then for $J = M_H$ and for all $t \geqslant 1$ we have $w_t/t \in \mathcal{W}$ and,*

$$\sup_{\bar{w} \in \mathcal{W}} \bar{w} J \tfrac{w_t}{t} \leqslant \delta + \tfrac{\rho}{t}.$$

In particular, in $\frac{\rho}{\varepsilon}$ iterations, Algorithm 5 obtain $(\delta + \varepsilon)$-solution of the saddle point problem (6.2)*.*

The analysis of this lemma closely follows the analysis of Nesterov's dual extrapolation.

Note that, each iteration consists of $O(1)$ matrix-vector multiplications, $O(1)$ vector additions, and $O(1)$ calls to the approximate oracle. Since the former two are parallelizable to $O(\log n)$ depth, the same remains to be shown for the oracle computation to complete the proof of the run-time in Theorem 6.1.1.

Recall from the discussion in Section 6.3 that the critical bottleneck of Nesterov's method is that diameter of the ℓ_∞ ball is $\Omega(\sqrt{n})$, which is achieved even in the Euclidean ℓ_2 norm. This makes ρ in Lemma 6.4.7 to also be $\Omega(\sqrt{n})$, which can be a major bottleneck for high dimensional LPs, which are commonplace among real-world applications.

Although, on the face of it, area convexity applied to the saddle point formulation (6.2) has a similar framework to Nesterov's dual extrapolation, the challenge is to construct a ϕ for which we can overcome the above bottleneck. Particularly, there are three key challenges to tackle:

1. We need to show that existence of a function ϕ that is area convex with respect to H on $\mathcal{W}$.
2. $\phi : \mathcal{W} \to [-\rho, 0]$ should be such that ρ is not too large.
3. There should exist an efficient δ-OSO for ϕ.

In the next subsection, we focus on these three aspects in order to complete our analysis.

6.4.3 Choosing an area convex function

First, we consider a simple 2-D gadget function and prove a "nice" property of this gadget. Using this gadget, we construct a function which can be shown to be area convex using the aforementioned property of the gadget.

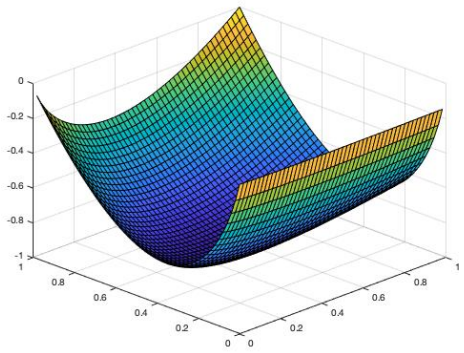
Let $\gamma_\beta : \mathbb{R}^2_+ \to \mathbb{R}$ be a function parameterized by β defined as

$$\gamma_\beta(a, b) = ba \log a + \beta b \log b.$$

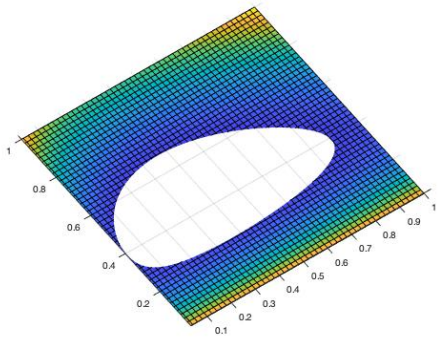
Lemma 6.4.8 *Suppose* $\beta \geqslant 2$. *Then* $d^2\gamma_\beta(a, b) \succeq \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ *for all* $a \in (0, 1]$ *and* $b > 0$.

We note in Figure 6.1 that the function γ_β is indeed convex. However, its level curves become straight near the boundary implying that this function is not strongly convex.

Now, using the function γ_β, we construct a function ϕ and use the sufficiency criterion provided in Lemma 6.4.5 to show that ϕ is area convex with respect to J on $\mathcal{W}$. Note that our set of interest $\mathcal{W}$ is not full-dimensional, whereas Lemma (6.4.5) is only stated for int and not for relint. To get around this difficulty, we consider a larger set $\overline{\mathcal{W}} \supset \mathcal{W}$ such that $\overline{\mathcal{W}}$ is full dimensional and ϕ is area convex on $\overline{\mathcal{W}}$. Then we use Remark 6.4.3 to obtain the final result, i.e., area convexity of ϕ.



(a) Auxiliary view



(b) Sublevel set $\gamma(x, y) \leqslant -0.5$

Figure 6.1: Sublevel set for area convex function γ_β.

Theorem 6.4.9 *Let* $w = (x, u, y, z)$ *and define*

$\phi(w) := \sum_{i=1}^{p} \sum_{j=1}^{n} P_{ij} \gamma_{p_i}(x_j, y_i) + \sum_{i=1}^{p} \gamma_2(u, y_i) + \sum_{i=1}^{c} \sum_{j=1}^{n} C_{ij} \gamma_{c_i}(x_j, z_i) + \sum_{i=1}^{c} \gamma_2(u, z_i)$,

where $p_i = 2 * \frac{\|P\|_\infty}{\|P_{i,:}\|_1}$ *and* $c_i = 2 * \frac{\|C\|_\infty}{\|C_{i,:}\|_1}$, *then* ϕ *is area convex with respect to* $\frac{1}{3} \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix}$ *on set* $\overline{\mathcal{W}} := \mathcal{B}^{n+1}_{+,\infty}(1) \times \Delta^+_p \times \Delta^+_c$. *In particular, it also implies* $6\sqrt{3}\phi$ *is area convex with respect to* $2\sqrt{3} \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix}$ *on set* $\mathcal{W}$.

Theorem 6.4.9 addresses the first part of the key three challenges. Next, Lemma 6.4.10 shows an upper bound on the range of ϕ.

Lemma 6.4.10 *Function* $\phi : \mathcal{W} \to [-\rho, 0]$ *then* $\rho = O(\|P\|_\infty \log p + \|C\|_\infty \log c)$.

Finally, we need an efficient δ-OSO. Consider the following alternating minimization algorithm.

Algorithm 6 δ-OSO for ϕ

Input $a \in \mathbb{R}^{n+1}, a^1 \in \mathbb{R}^p, a^2 \in \mathbb{R}^c, \delta > 0$

Initialize $(x^0, u^0) \in \mathcal{B}^n_{+,\infty} \times \{1\}$ arbitrarily.

for $k = 1, \ldots, K$ **do**

$$(y^k, z^k) \leftarrow \operatorname*{argmax}_{y \in \Delta^+_c,\ z \in \Delta^+_p} y^T a^1 + z^T a^2 - \phi(x^{k-1}, u^{k-1}, y, z)$$

$$(x^k, u^k) \leftarrow \operatorname*{argmax}_{(x,u) \in \mathcal{B}^n_{+,\infty} \times \{1\}} [x^T\ u]a - \phi(x, u, y^k, z^k)$$

end for

[10] shows the following convergence result.

Lemma 6.4.11 *For $\delta > 0$, Algorithm 6 is a δ-OSO for ϕ which converges in $O(\log \frac{1}{\delta})$ iterations.*

We show that for our chosen ϕ, we can perform the two argmax computations in each iteration of Algorithm 6 analytically in time $O(\mathrm{nnz}(P) + \mathrm{nnz}(C))$, and hence we obtain a δ-OSO which takes $O((\mathrm{nnz}(P) + \mathrm{nnz}(C) \log \frac{1}{\delta})$ total work. Parallelizing matrix-vector multiplications eliminates the dependence on $\mathrm{nnz}(P)$ and $\mathrm{nnz}(C)$, at the cost of another $\log(N)$ term.

Lemma 6.4.12 *Each* argmax *in Algorithm 6 can be computed as follows:*

$x^k = \min\{\exp\big(\frac{a}{P^T y^k + C^T z^k} - 1\big), \mathbf{1}_n\}$ *for all* $j \in [n]$.

$y^k = \mathrm{proj}_{\Delta^+_p}\big(\exp\{\frac{1}{2(\|P\|^+_\infty 1)}(a^1 - Px^{k-1} \log x^{k-1})\}\big)$

$z^k = \mathrm{proj}_{\Delta^+_c}\big(\exp\{\frac{1}{2(\|C\|^+_\infty 1)}(a^2 - Cx^{k-1} \log x^{k-1})\}\big)$

In particular, we can compute x^k, y^k, z^k in $O(\mathrm{nnz}(P) + \mathrm{nnz}(C))$ work and $O(\log N)$ parallel time.

As a result of the above lemma, we obtain that three key challenges are overcome due the area convex regularization and hence, we obtain convergence to an ε-solution of MPC at the rate $\widetilde{O}(wN\varepsilon^{-1})$.

6.5 Proof of auxiliary results

In this section, we include proofs of lemmas from the main paper. In some cases, the lemmas are direct restatements of results from other papers, for which we provide appropriate pointers.

6.5.1 Proof of Lemma 6.3.1

Consider an arbitrary strongly convex function d. Assume WLOG that $d(0) = 0$. (otherwise, we can shift it accordingly). We will show that $\max_{x \in \mathcal{B}^n_\infty(r)} d(x) \geqslant \frac{nr^2}{2}$ by induction on n for set $\mathcal{B}^n_\infty(r)$. This suffices because $\mathcal{B}^n_{+,\infty}(1)$ is isomorphic to $\mathcal{B}^n_\infty(\frac{1}{2})$. The claim holds for $n = 1$ by the definition of strong convexity. Now, suppose it is true for $n-1$. Then there exists $\bar{x} \in \mathcal{B}^{n-1}_\infty(r)$ such that $d(\bar{x}) \geqslant \frac{(n-1)r^2}{2}$. Moving r units in the last coordinate from $\bar{x}$ in the direction of nonnegative slope, suppose we reach $\widehat{x} \in \mathcal{B}^n_\infty(r)$. Then, due to strong convexity of d, we have $d(\widehat{x}) \geqslant d(\bar{x}) + \frac{1}{2}\|\widehat{x} - \bar{x}\|^2_\infty \geqslant \frac{(n-1)r^2}{2} + \frac{r^2}{2} = \frac{nr^2}{2}$.

6.5.2 Proof of Lemma 6.4.2

Suppose we are given (x, y, z) such that $\sup_{(\bar{x},\bar{y},\bar{z}) \in \mathcal{B}^n_{+,\infty} \times \Delta^+_p \times \Delta^+_c} L(x, \bar{y}, \bar{z}) - L(\bar{x}, y, z) \leqslant \varepsilon$. If there exists $\widetilde{x}$ which is feasible for MPC then choosing $\bar{x} = \widetilde{x}$ then $L(\widetilde{x}, y, z) \leqslant 0$. Hence we have

$$\begin{aligned} &\sup_{(\bar{y},\bar{z}) \in \Delta^+_p \times \Delta^+_c} L(x, \bar{y}, \bar{z}) \leqslant \varepsilon \\ \Rightarrow &\|[Px - \mathbf{1}_p]_+\|_\infty + \|[-Cx + \mathbf{1}_c]_+\|_\infty \leqslant \varepsilon, \end{aligned}$$

where implication follows by optimality over extended simplices Δ^+_p, Δ^+_c. So we obtain, if there exist a feasible solution for MPC then x is ε-approximate solution of MPC.

On the other hand, suppose x is not an ε-approximate solution. Then

$$\begin{aligned} &\max\{\|[Px - \mathbf{1}_p]_+\|_\infty, \|[-Cx + \mathbf{1}_c]_+\|_\infty\} > \varepsilon \\ \Rightarrow \sup_{(\bar{y},\bar{z}) \in \Delta^+_p \times \Delta^+_c} &L(x, \bar{y}, \bar{z}) = \|[Px - \mathbf{1}_p]_+\|_\infty + \|[-Cx + \mathbf{1}_c]_+\|_\infty > \varepsilon \end{aligned}$$

Let $(\widehat{y}, \widehat{z}) \in \Delta_p^+ \times \Delta_c^+$ such that $L(x, \widehat{y}, \widehat{z}) > \varepsilon$ then we have

$$\sup_{\bar{x} \in \mathcal{B}^n_{+,\infty}} L(x, \widehat{y}, \widehat{z}) - L(\bar{x}, y, z) \leqslant \varepsilon$$

$$\Rightarrow L(x, \widehat{y}, \widehat{z}) - \inf_{\bar{x} \in \mathcal{B}^n_{+,\infty}} L(\bar{x}, y, z) \leqslant \varepsilon$$

$$\Rightarrow \inf_{\bar{x} \in \mathcal{B}^n_{+,\infty}} L(\bar{x}, y, z) > 0$$

Hence, if x is not ε-approximate solution of MPC then (y, z) satisfy $y^T(P\bar{x} - \mathbf{1}_p) + z^T(-C\bar{x} + \mathbf{1}_c) > 0$ for all $\bar{x} \in \mathcal{B}^n_{+,\infty}(1)$ implying that MPC is infeasible.

6.5.3 Proof of Lemma 6.4.4

Let $B = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ and $T := \begin{bmatrix} A & -B \\ B & A \end{bmatrix}$.
Then $A \succeq_{\mathrm{i}} B$ iff $T \succeq 0$ iff all principle minors of T are nonnegative. Now, $T \succeq 0$ implies $A \succeq 0$. It is easy to verify that third principle minor is nonnegative iff $\det(A) \geqslant 1$. So $T \succeq 0$ implies A must be invertible. Then, applying Schur complement lemma, we obtain that $T \succeq 0 \Leftrightarrow A + BA^{-1}B \succeq 0$. Now let $A = \begin{bmatrix} a & b \\ b & d \end{bmatrix}$ then $A^{-1} = \frac{1}{ad-b^2} \begin{bmatrix} d & -b \\ -b & a \end{bmatrix}$. It is easy to verify that $A + BA^{-1}B = A(1 - \frac{1}{\det(A)})$. This implies $T \succeq 0 \Leftrightarrow A \succeq 0$ and $\det(A) \geqslant 1$. Hence we conclude the proof.

6.5.4 Proof of Lemma 6.4.5

This lemma appears exactly as Theorem 1.6 in [100]. The proof follows from the same.

6.5.5 Proof of Proposition 6.4.6

1.

$$
\begin{aligned}
A \succeq_{\mathrm{i}} B &\Leftrightarrow \begin{bmatrix} A & -B \\ B & A \end{bmatrix} \geq 0 \\
&\Leftrightarrow x^T A x + y^T A y + y^T B x - x^T B y \geqslant 0, \quad \forall\, x, y \\
&\Leftrightarrow x^T A x + y^T A y - y^T B x + x^T B y \geqslant 0, \quad \forall\, x, y \\
&\Leftrightarrow \begin{bmatrix} A & B \\ -B & A \end{bmatrix} \geq 0 \Leftrightarrow A \succeq_{\mathrm{i}} (-B)
\end{aligned}
$$

Here, the third equivalence follows after replacing y by $-y$. Hence we conclude the proof of part 1.

2.

$$
A \succeq_{\mathrm{i}} B \Leftrightarrow \begin{bmatrix} A & -B \\ B & A \end{bmatrix} \geq 0 \Rightarrow \begin{bmatrix} \lambda A & -\lambda B \\ \lambda B & \lambda A \end{bmatrix} \geq 0 \Leftrightarrow \lambda A \geq \lambda B
$$

3. $A \succeq_{\mathrm{i}} B$ implies $\begin{bmatrix} A & -B \\ B & A \end{bmatrix} \geq 0$. Similarly $C \succeq_{\mathrm{i}} D$ implies $\begin{bmatrix} C & -D \\ D & C \end{bmatrix} \geq 0$. Hence

$$
\begin{bmatrix} A + C & -(B + D) \\ (B + D) & (A + C) \end{bmatrix} \geq 0.
$$

So we obtain $A + C \succeq_{\mathrm{i}} (B + D)$.

6.5.6 Proof of Lemma 6.4.7

This lemma appears as Theorem 1.3 in [100], and the proof follows from the same.

6.5.7 Proof of Lemma 6.4.8

We use equivalent characterization proved in Lemma 6.4.4. We need to show that $d^2\gamma_\beta(a,b) \succeq 0$ and $\det(d^2\gamma_\beta(a,b)) \geqslant 1$ for all $a \in (0,1]$ and $b > 0$. First of all, note that $d^2\gamma_\beta$ is well-defined on this domain. In particular, we can write

$$d^2\gamma_\beta(a,b) = \begin{bmatrix} \frac{\beta}{b} & 1+\log a \\ 1+\log a & \frac{b}{a} \end{bmatrix}.$$

Note that a 2×2 matrix is PSD if and only if its diagonal entries and determinant are nonnegative. Clearly diagonal entries of $d^2\gamma_\beta(a,b)$ are nonnegative for the given values of β, a and b. Hence, in order to prove the lemma, it suffices to show that $\det(d^2\gamma_\beta(a,b)) \geqslant 1$.

$\det(d^2\gamma_\beta(a,b)) = \frac{\beta}{a} - (1+\log a)^2$ is only a function of a for any fixed value of $\beta \geqslant 2$. Moreover, it can be shown that $\det(d^2\gamma_\beta)$ is a decreasing function of a on set $(0,1]$. Clearly, the minimum occurs at $a = 1$. However, $\det(d^2\gamma_\beta(1,b)) = \beta - 1 \geqslant 1$ for all $b > 0$. Hence we have that $\det(d^2\gamma_\beta(a,b)) \geqslant 1$ for all $a \in (0,1], b > 0$ and $\beta \geqslant 2$.

Finally to see the claim that $\det(d^2\gamma_\beta)$ is a decreasing function of $a \in (0,1]$ for any $\beta \geqslant 2$, consider

$$\begin{aligned} \frac{d}{da}\big(\det(d^2\gamma_\beta(a,b))\big) &= -\frac{\beta}{a^2} - \frac{2(1+\log a)}{a} \\ &\leqslant -\frac{2(1+a(1+\log a))}{a^2} < 0 \end{aligned}$$

where the last inequality follows from the observation that $1 + a + a\log a > 0$ for all $a \in (0,1]$. Hence we conclude the proof.

6.5.8 Proof of Theorem 6.4.9

Note that $\gamma_{c_i}, \gamma_{p_i}$ are twice differentiable in the $\mathrm{int}(\overline{\mathcal{W}})$. So by Lemma 6.4.5 part 2, it is sufficient to prove that $d^2\phi(w) \succeq_{\mathrm{i}} J$ for all $w \in \mathrm{int}(\overline{\mathcal{W}})$.

By definition, we have $\gamma_{c_i} \geqslant 2$ for all $i \in [c]$ and $\gamma_{p_i} \geqslant 2$ for all $i \in [p]$. Moreover $x_j \in (0,1)$ and $y_i > 0, z_i > 0$ for any $w = (x,u,y,z) \in \operatorname{int}(\mathcal{W})$. Then by Lemma 6.4.8 and Proposition 6.4.6, we have

$$\begin{aligned} d^2\phi(w) &= \sum_{i=1}^{p}\sum_{j=1}^{n} P_{ij} d^2\gamma_{p_i}(x_j, y_i) + \sum_{i=1}^{p} d^2\gamma_2(u, y_i) + \sum_{i=1}^{c}\sum_{j=1}^{n} C_{ij} d^2\gamma_{c_i}(x_j, y_i) + \sum_{i=1}^{c} d^2\gamma_2(u, z_i) \\ &\succeq_{\mathrm{i}} \Big(\sum_{i=1}^{p}\sum_{j=1}^{n} -P_{ij} e_j \otimes e_{n+1+i} + \sum_{i=1}^{p} e_{n+1} \otimes e_{n+1+i} \\ &\qquad + \sum_{i=1}^{c}\sum_{j=1}^{n} C_{ij} e_j \otimes e_{n+p+i} + \sum_{i=1}^{c} (-1) e_{n+1} \otimes e_{n+1+p+i} \Big), \end{aligned} \tag{6.4}$$

where $e_k \otimes e_l = e_k e_l^T - e_l e_k^T$. Here we used $P_{ij} d^2\gamma_{p_i}(x_j, y_i) \succeq_{\mathrm{i}} -P_{ij} e_j \otimes e_{n+1+i}$ using Lemma 6.4.8, Proposition 6.4.6 part 1, part 2 and $C_{ij} d^2\gamma_{c_i}(x_j, y_i) \succeq_{\mathrm{i}} C_{ij} e_j \otimes e_{n+1+p+i}$ using Lemma 6.4.8, Proposition 6.4.6 part 2. Similar arguments can be made about terms inside the other two summations. Finally we used Proposition 6.4.6 part 3 to obtain (6.4). Note matrix in the last sum term is in fact J.

It is clear that since $d^2\phi \succeq_{\mathrm{i}} J$ hence using Proposition 6.4.6 part 2, we have $d^2 6\sqrt{3}\phi \succeq_{\mathrm{i}} 6\sqrt{3} J$. Then by Lemma 6.4.5 part 2, we obtain $6\sqrt{3}\phi$ is area convex with respect to $2\sqrt{3} \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix}$ on set $\overline{\mathcal{W}}$.

Note that the set of interest $\mathcal{W} \subset \overline{\mathcal{W}}$. Moreover, $\mathcal{W}$ is a convex subset. By Remark 6.4.3, one can see that $6\sqrt{3}\phi$ is area convex with respect to $2\sqrt{3} \begin{bmatrix} P & -\mathbf{1}_p \\ -C & \mathbf{1}_c \end{bmatrix}$ on set $\mathcal{W}$. Hence we conclude the proof.

6.5.9 Proof of Lemma 6.4.10

Note that $\gamma_\beta(a,b) \leqslant 0$ for any $a \in [0,1], b \in [0,1], \beta \geqslant 0$. Since $P_{ij} \geqslant 0, C_{kj} \geqslant 0$ for all possible values of i, j, k hence we clearly have $\phi(w) \leqslant 0$ for all $w \in \mathcal{W}$. Now we prove that lower bound is not too small.

We have

$$\begin{aligned}
\sum_{i=1}^{p}\sum_{j=1}^{n} P_{ij}\gamma_{p_i}(x_j, y_i) &= \sum_{i=1}^{p}\sum_{j=1}^{n} P_{ij}(y_i x_j \log x_j + p_i y_i \log y_i) \\
&\geqslant -\sum_{i=1}^{p}\sum_{j=1}^{n} P_{ij} y_i \frac{1}{e} + \sum_{i=1}^{p} p_i y_i \log y_i \sum_{j=1}^{p} P_{ij} \\
&= -\sum_{i=1}^{p}\sum_{j=1}^{n} P_{ij} y_i \frac{1}{e} + \sum_{i=1}^{p} 2\|P\|_\infty y_i \log y_i \\
&\geqslant -\sum_{i=1}^{p} \frac{\|P\|_\infty}{e} y_i + \sum_{i=1}^{p} 2\|P\|_\infty y_i \log y_i \\
&\geqslant -\frac{\|P\|_\infty}{e} - 2\|P\|_\infty \log p
\end{aligned}$$

Note that if $w \in \mathcal{W}$ implies $u = 1$. So

$$\sum_{i=1}^{p} \gamma_2(u, y_i) = \sum_{i=1}^{p} 2y_i \log(y_i) \geqslant -2 \log p$$

Similarly, we have

$$\begin{aligned}
\sum_{i=1}^{c}\sum_{j=1}^{n} C_{ij}\gamma_{c_i}(x_j, z_i) &\geqslant -\frac{\|C\|_\infty}{e} - 2\|C\|_\infty \log c \\
\sum_{i=1}^{c} \gamma_2(u, z_i) &\geqslant -2 \log c
\end{aligned}$$

Taking sum of all four terms, we conclude the proof.

6.5.10 Proof of Lemma 6.4.12

Note that maximization with respect to u is trivial since $u = 1$ is a fixed variable. We first look at maximization with respect to $x \in \mathcal{B}^n_{+,\infty}(1)$. Writing the first order necessary condition of Lagrange

multipliers, we have

$$a_j - \sum_{i=1}^{p} P_{ij} \left.\frac{\partial}{\partial t}\gamma_{p_i}(t,v)\right|_{(t,v)=(x_j,y_i)} - \sum_{i=1}^{c} C_{ij} \left.\frac{\partial}{\partial t}\gamma_{c_i}(t,v)\right|_{(t,v)=(x_j,z_i)} - \lambda_j = 0$$
$$\Rightarrow a_j - \Big\{\sum_{i=1}^{p} P_{ij}y_i + \sum_{i=1}^{c} C_{ij}z_i\Big\}(1+\log x_j) - \lambda_j = 0.$$

Here λ_j is the Lagrange multiplier corresponding to the case that $x_j = 1$. By complimentary slackness, we have $\lambda_j > 0$ iff $x_j = 1$.

This implies $x_j = \min\left\{\exp\left(\frac{a_j}{\sum_{i=1}^{p} P_{ij}y_i + \sum_{i=1}^{c} C_{ij}z_i} - 1\right), 1\right\}$ for all $j \in [n]$.

Now we consider maximization with respect to y, z. Note that there are no cross-terms of y_i and z_i, i.e., $\frac{\partial \gamma_{p_i}}{\partial y_i}$ is independent of z variable and vice-versa. So we can optimize them separately. From first order necessary condition of Lagrange multipliers for y, we have

$$a_i^1 - \sum_{j=1}^{n} P_{ij} \left.\frac{\partial}{\partial v}\gamma_{p_i}(t,v)\right|_{(t,v)=(x_j,y_i)} - \left.\frac{\partial}{\partial v}\gamma_2(t,v)\right|_{(t,v)=(u,y_i)} - \lambda = 0$$
$$\Rightarrow a_i^1 - \sum_{j=1}^{n} P_{ij}(x_j \log x_j + p_i(1+\log y_i)) - u\log u\big|_{u=1} - 2(1+\log y_i) - \lambda = 0$$
$$\Rightarrow a_i^1 - \sum_{j=1}^{n} P_{ij}x_j \log x_j - 2(\|P\|_\infty^+ 1)(1+\log y_i) - \lambda = 0$$

where last relation follows due to definition of p_i and λ is Lagrange multiplier corresponding to the constraint $\sum_{i=1}^{p} y_i \leqslant 1$. By comple mentary slackness, we have $\lambda > 0$ iff $\sum_{i=1}^{p} y_i = 1$.

Eliminating λ from above equations, we obtain $y = \mathrm{proj}_{\Delta_p^+}\left(\exp\left\{\frac{1}{2(\|P\|_\infty^+ 1)}(a^1 - Px\log x)\right\}\right)$.

Similarly, we obtain $z = \mathrm{proj}_{\Delta_c^+}\left(\exp\left\{\frac{1}{2(\|C\|_\infty^+ 1)}(a^2 - Cx\log x)\right\}\right)$.

It is clear from the analytical expressions that for each iteration of Algorithm 6, we need $O(\mathrm{nnz}(P) + \mathrm{nnz}(C))$ time. Hence total runtime of Algorithm 6 is $O((\mathrm{nnz}(P) + \mathrm{nnz}(C))\log\frac{1}{\delta})$.

6.5.11 Proof of width reduction for the MPC problem

In Section 6.3, we made the assumption that all entries

This assumption follows from the results in [107]. We outline this proof in this section for completeness.

For the purpose of this proof, we introduce notation $[k] := \{1, \ldots, k\}$.

Suppose we are given an instance of mixed packing covering of the form

$$Px \leqslant \mathbf{1}_p, Cx \geqslant \mathbf{1}_c, x \geqslant \mathbf{0}_n. \tag{6.5}$$

Case 1: For each column $P_{:,i}$ associated with variable x_i, let $P_{j_i,i} := \max_{j\in[p]} P_{ji} > 0$. Then we consider the following updates to MPC in order to reduce diameter.

Suppose, without loss of generality, $C_{1,i} = \max_{j\in[c]} C_{ji}$ and $C_{ci} = \min_{j\in[c]} C_{ji}$. If $C_{1i} \leqslant P_{j_i,i}$ then we can update $\bar{P}_{:,i} = \frac{1}{P_{j_i,i}} P_{:,i}$, $\bar{C}_{:,i} = \frac{1}{P_{j_i,i}} C_{:,i}$ and $\bar{x}_i = P_{j_i,i} x_i$. Then we observe that each element in $\bar{P}_{:,i}, \bar{C}_{:,i}$ is at most 1. Moreover, due to the packing constraint $\bar{P}_{j_i,:}\bar{x} \leqslant 1$, we note that for any feasible $\bar{x}$, $\bar{P}_{j_i,i}\bar{x}_i \leqslant 1$. Finally, since $\bar{P}_{j_i,i} = 1$, we have that $\bar{x}_i \leqslant 1$ lies in the support of constraint set. So we replaced the i-th column and corresponding i-th variable of the system by an equivalent system.

Similarly, if $C_{c,i} \geqslant P_{j_i,i}$ then consider x^{sol} defined as

$$x^{sol}_k := \begin{cases} \frac{1}{P_{j_i,i}} & \text{if } k = i \\ 0 & \text{otherwise.} \end{cases}$$

Then x^{sol} is already a feasible solution of MPC. So we may assume that $C_{ci} < P_{j_i,i} < C_{1i}$. In this case, define $r_i = \frac{C_{1i}}{P_{j_i,i}}$ and $n_i = \lceil \log r_i \rceil$. We make n_i copies of the column $C{:}, i$ and denote by the tuple (i, l) the columns of a new matrix $\widehat{C}_{:,(i,l)}$ where $l \in [n_i]$. Similarly, we add n_i copies of variable x_i, denoted as $\widehat{x}_{(i,l)}$. We make similar changes to $P_{:,i}$. Note that this system is equivalent to earlier system in the sense that any solution $\widehat{x}_{(i,l)}, l \in [n_i]$ can be converted into a solution of the earlier system since $x_i = \sum_{l\in[n_i]} \widehat{x}_{(i,l)}$. However, this allows us to reduce the elements of $\widehat{C}$ along

with certain box constraints on $\widehat{x}_i$, which was our original goal. For each $j \in [c], l \in [n_i]$, redefine

$$\widehat{C}_{j,(i,l)} = \min\{C_{ji}, 2^l P_{j_i,i}\}$$

and for variable $\widehat{x}_{(i,l)}$, add the constraint

$$\widehat{x}_{i,l} \leqslant \frac{2}{2^l P_{j_i,i}}. \tag{6.6}$$

Claim 6.5.1 *MPC* (6.5) *and the new system defined by matrices $\widehat{C}, \widehat{P}$ and variable $\widehat{x}$ are equivalent.*

Proof. For this proof, let us focus on i-th column and i-th variable.

For any feasible solution $\widehat{x}$, consider $x_i = \sum_{l \in [n_i]} \widehat{x}_{i,l}$. This x_i does not violate any covering constraint since $\widehat{C}_{j,(i,l)} \leqslant C_{ji}$. The packing constraints also follow because we have not made any changes to the elements corresponding to the packing constraints $\widehat{P}_{j,(i,l)}$.

For the other direction, the key fact to note is that any feasible x satisfies $x_i \leqslant \frac{1}{P_{j_i,i}}$ due to packing constraint $P_{j_i,:}x \leqslant 1$. Let l_i be the largest index such that

$$x_i \leqslant \frac{2}{2^{l_i} P_{j_i,i}},$$

and then let

$$\widehat{x}_{(i,l)} = \begin{cases} x_i & \text{if } l = l_i \\ 0 & \text{otherwise.} \end{cases}$$

By construction, $\widehat{x}_{(i,l)}$ satisfies the constraint in (6.6) for all $l \in [n_i]$. Moreover, for constraint j, we must have $\widehat{C}_{j,:}\widehat{x} \geqslant 1$. Note that if $\widehat{C}_{j,(i,l_i)} = C_{ji}$ then there is nothing to prove. So we assume that $C_{ji} > \widehat{C}_{j,(i,l_i)} = 2^{l_i} P_{j_i,i}$. Then we must have that $l_i < n_i$ in this case, by definition of n_i. This then gives $\widehat{x}_{(i,l_i)} = x_i \geqslant \frac{1}{2^{l_i} P_{j_i,i}}$ by our choice of l_i being the largest possible. Then we know that

$\widehat{C}_{j,(i,l_i)} = 2^{l_i} P_{j_i,i}$, and hence the j-th covering constraint is satisfied.

Packing constraints are satisfied trivially since there is no change in elements of $\widehat{P}_{:,(i,l)}$ for all $l \in [n_i]$. Hence the claim follows. □

Finally the proof follows by change of variables as $\overline{x}_{(i,l)} = 2^{l-1} P_{j_i,i}$ and $\overline{C}_{:,(i,l)} = \frac{1}{2^{l-1} P_{j_i,i}} \widehat{C}_{:,(i,l)}$. Further, note that all elements of $\overline{P}_{:,(i,l)}$ are at most 1 for all $l \in [n_i]$, and all elements of $\overline{C}_{:,(i,l)}$ are at most 2 for all $l \in [n_i]$ and $\overline{x}_{i,l} \leqslant 1$ for all $l \in [n_i]$.

Case 2: Suppose $P_{j_i,i} = 0$.. This implies that in variable x_i, this is a purely covering problem. So we can increase x_i to satisfy the jth covering constraint such that $C_{ji} > 0$ independent of the packing constraints and problem reduces to smaller packing covering problem in remaining variables and covering constraints j such that $C_{ji} = 0$. For this smaller packing covering problem, we can apply the method in Case 1 again.

6.5.12 Application to the Densest Subgraph problem

In this section, we apply the result in Theorem 6.1.1 to the *densest subgraph problem*.

We define the density of a graph $G = \langle V, E \rangle$ as $|V|/|E|$ (half the average degree of G). Hence, the densest subgraph of G is induced on a subset of vertices $U \subseteq V$ such that

$$U := \operatorname*{argmax}_{S \subseteq V} \frac{|E(S)|}{|S|},$$

where $E(S)$ denotes the set of edges in the subgraph of G induced by S.

The following is a well-known LP formulation of the densest subgraph problem, introduced in [23], which we denote using $\textsc{Primal}(G)$. The optimal objective value is known to be ρ^*_G.

$$
\begin{aligned}
\text{maximize} \quad & \sum_{e \in E} y_e \\
\text{subject to} \quad & y_e \leqslant x_u, x_v, \qquad \forall e = uv \in E \\
& \sum_{v \in V} x_v \leqslant 1, \\
& y_e \geqslant 0, x_v \geqslant 0, \quad \forall e \in E, \forall v \in V
\end{aligned}
$$

We then construct the dual LP for the above problem. Let $f_e(u)$ be the dual variable associated with the first $2m$ constraints of the form $y_e \leqslant x_u$, and let D be associated with the last constraint. We get the following LP, which we denote by $\textsc{Dual}(G)$, and whose optimum is also ρ^*_G.

$$
\begin{aligned}
\text{minimize} \quad & D \\
\text{subject to} \quad & f_e(u) + f_e(v) \geqslant 1, && \forall e = uv \in E \\
& \sum_{e \ni v} f_e(v) \leqslant D, && \forall v \in V \\
& f_e(u) \geqslant 0, f_e(v) \geqslant 0, && \forall e = uv \in E
\end{aligned}
$$

Parametrizing with respect to D, this becomes a mixed packing covering LP. The solution to the densest subgraph problem is simply the smallest value of D for which the LP is feasible. Since D can take at most $O(|V||E|) \leqslant O(|V|^3)$ values in total, the densest subgraph problem can be reduced to solving $O(\log |V|)$ instances of MPC, where the number of nonzeros N in the matrix is $O(|E|)$ and the width w is simply the maximum degree in G. This gives the following corollary.

Corollary 6.5.2 *Given a graph $G = \langle V, E \rangle$ with maximum degree Δ, we can find the $(1+\varepsilon)$-approximation to the maximum subgraph density of G, ρ^*_G, in parallel time $\widetilde{O}(\Delta \varepsilon^{-1})$ and total work $\widetilde{O}(\Delta |E| \varepsilon^{-1})$.*

The previous fastest algorithms for densest subgraph do not depend on Δ - however, their dependence on $1/\varepsilon$ is quadratic [7]. Corollary 6.5.2 gives the fastest algorithm for this problem in the high precision regime ($\varepsilon < 1/\Delta$), since its dependence

REFERENCES

[1] Z. Allen-Zhu and E. Hazan, "Variance reduction for faster non-convex optimization," *International Conference on Machine Learning*, pp. 699–707, 2016.

[2] Z. Allen-Zhu and L. Orecchia, "Nearly linear-time packing and covering LP solvers - achieving width-independence and -convergence," *Math. Program.*, vol. 175, no. 1-2, pp. 307–353, 2019.

[3] R. Andreani, G. Haeser, and J. M. Martínez, "On sequential optimality conditions for smooth constrained optimization," *Optimization*, vol. 60, no. 5, pp. 627–641, 2011.

[4] R. Andreani, J. M. Martínez, A. Ramos, and P. J. S. Silva, "Strict constraint qualifications and sequential optimality conditions for constrained optimization," *Mathematics of Operations Research*, vol. 43, pp. 693–717, 2018.

[5] A. Y. Aravkin, J. V. Burke, D. Drusvyatskiy, M. P. Friedlander, and S. Roy, "Level-set methods for convex optimization," *Mathematical Programming*, pp. 1–32, 2018.

[6] R. Arora, A. Basu, P. Mianjy, and A. Mukherjee, "Understanding deep neural networks with rectified linear units," 2016.

[7] B. Bahmani, A. Goel, and K. Munagala, "Efficient primal-dual graph algorithms for mapreduce," in *Algorithms and Models for the Web Graph - 11th International Workshop, WAW 2014, Beijing, China, December 17-18, 2014, Proceedings*, 2014, pp. 59–78.

[8] Y. Bartal, J. W. Byers, and D. Raz, "Fast, distributed approximation algorithms for positive linear programming with applications to flow control," *SIAM J. Comput.*, vol. 33, no. 6, pp. 1261–1279, 2004.

[9] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[10] A. Beck, "On the convergence of alternating minimization for convex programming with applications to iteratively reweighted least squares and decomposition schemes," *SIAM Journal on Optimization*, vol. 25, no. 1, pp. 185–209, 2015.

[11] A. Ben-Tal and A. Nemirovski, "Non-euclidean restricted memory level method for large-scale convex optimization," *Mathematical Programming*, vol. 102, pp. 407–456, 2005.

[12] D. P. Bertsekas, *Nonlinear programming*. Athena Scientific, 1999.

[13] D. P. Bertsekas, *Convex optimization algorithms*. Athena Scientific Belmont, 2015.

[14] D. Bienstock and G. Iyengar, "Approximating fractional packings and coverings in o(1/epsilon) iterations," *SIAM J. Comput.*, vol. 35, no. 4, pp. 825–854, 2006.

[15] A. Blum and R. L. Rivest, "Training a 3-node neural network is np-complete," in *Proceedings of the First Annual Workshop on Computational Learning Theory*, ser. COLT '88, 1988, pp. 9–18.

[16] D. Boob, Q. Deng, and G. Lan, "Stochastic first-order methods for convex and nonconvex functional constrained optimization," *arXiv preprint arXiv:1908.02734*, 2019.

[17] P. Bradley and O. L. Mangasarian, "Feature selection via concave minimization and support vector machines," in *Proceedings of International Conference on Machine Learning (ICML'98)*, Morgan Kaufmann, 1998, pp. 82–90.

[18] S. Bubeck, "Theory of convex optimization for machine learning," *arXiv preprint arXiv:1405.4980*, vol. 15, 2014.

[19] E. J. Candès, Y. Plan, *et al.*, "Near-ideal model selection by ℓ_1 minimization," *The Annals of Statistics*, vol. 37, no. 5A, pp. 2145–2177, 2009.

[20] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted l1 minimization," *arXiv preprint arXiv:0711.1612*, 2007.

[21] C. Cartis, N. I. Gould, and P. L. Toint, "On the complexity of finding first-order critical points in constrained nonlinear optimization," *Mathematical Programming*, vol. 144, no. 1, pp. 93–106, 2014.

[22] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.

[23] M. Charikar, "Greedy approximation algorithms for finding dense components in a graph," in *Proceedings of the Third International Workshop on Approximation Algorithms for Combinatorial Optimization*, ser. APPROX '00, Berlin, Heidelberg, 2000, pp. 84–95, ISBN: 3-540-67996-0.

[24] Y. Chen, G. Lan, and Y. Ouyang, "Optimal primal-dual methods for a class of saddle point problems," *SIAM Journal on Optimization*, vol. 24, no. 4, pp. 1779–1814, 2014.

[25] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th International Conference on Machine Learning*, ser. ICML '08, 2008, pp. 160–167.

[26] B. DasGupta, H. T. Siegelmann, and E. Sontag, "On a learnability question associated to neural networks with continuous activations," in *Proceedings of the Seventh Annual Conference on Computational Learning Theory*, ser. COLT '94, 1994, pp. 47–56.

[27] D. Davis and B. Grimmer, "Proximally guided stochastic subgradient method for nonsmooth, nonconvex problems," *arXiv preprint arXiv: 1707.03505v4*, 2017.

[28] S. S. Dey, G. Wang, and Y. Xie, *An approximation algorithm for training one-node relu neural network*, 2018. arXiv: `1810.03592 [math.OC]`.

[29] S. Diamond and S. Boyd, "Cvxpy: A python-embedded modeling language for convex optimization," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2909–2913, 2016.

[30] Q. T. Dinh, S. Gumussoy, W. Michiels, and M. Diehl, "Combining convex-concave decompositions and linearization approaches for solving bmis, with application to static output feedback," *arXiv preprint arXiv:1109.3320*, 2011.

[31] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the l1-ball for learning in high dimensions," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 272–279.

[32] H Edelsbrunner, J O'Rouke, and R Seidel, "Constructing arrangements of lines and hyperplanes with applications," *SIAM J. Comput.*, vol. 15, no. 2, pp. 341–363, 1986.

[33] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[34] F. Facchinei, V. Kungurtsev, L. Lampariello, and G. Scutari, "Ghost penalties in nonconvex constrained optimization: Diminishing stepsizes and iteration complexity," *arXiv preprint arXiv:1709.03384*, 2017.

[35] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American Statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001.

[36] C. Fang, C. J. Li, Z. Lin, and T. Zhang, "Spider: Near-optimal non- convex optimization via stochastic path-integrated differential estimator," *Advances in Neural Information Processing Systems*, pp. 687–697, 2018.

[37] R. Frostig, R. Ge, S. Kakade, and A. Sidford, "Un-regularizing: Approximate proximal point and faster stochastic algorithms for empirical risk minimization," in *International Conference on Machine Learning*, 2015, pp. 2540–2548.

[38] W. J. Fu, "Penalized regressions: The bridge versus the lasso," *Journal of Computational and Graphical Statistics*, vol. 7, no. 3, pp. 397–416, 1998.

[39] S. Ghadimi and G. Lan, "Stochastic first- and zeroth-order methods for nonconvex stochastic programming," *SIAM Journal on Optimization*, vol. 23(4), pp. 2341–2368, 2013.

[40] S. Ghadimi and G. Lan, "Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework," *SIAM Journal on Optimization*, vol. 22, no. 4, pp. 1469–1492, 2012.

[41] S. Ghadimi and G. Lan, "Accelerated gradient methods for nonconvex nonlinear and stochastic programming," *Mathematical Programming*, vol. 156, no. 1-2, pp. 59–99, 2016.

[42] P. Gong, C. Zhang, Z. Lu, J. Z. Huang, and J. Ye, "A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems," *International Conference on Machine Learning*, vol. 28, no. 2, pp. 37–45, 2013.

[43] J. ya Gotoh, A. Takeda, and K. Tono, "DC formulations and algorithms for sparse optimization problems," *Mathematical Programming*, vol. 169, no. 1, pp. 141–176, 2018.

[44] M. D. Grigoriadis and L. G. Khachiyan, "Approximate minimum-cost multicommodity flows in Õ(epsilon^{-2}knm) time," *Math. Program.*, vol. 75, pp. 477–482, 1996.

[45] O. Güler, "New proximal point algorithms for convex minimization," *SIAM Journal on Optimization*, vol. 2, no. 4, pp. 649–664, 1992.

[46] E. Y. Hamedani and N. S. Aybat, "A primal-dual algorithm for general convex-concave saddle point problems," *arXiv preprint arXiv:1803.01401*, 2018.

[47] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[48] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber, *Gradient flow in recurrent nets: The difficulty of learning long-term dependencies*, 2001.

[49] K. Khamaru and M. J. Wainwright, "Convergence guarantees for a class of non-convex and non-smooth optimization problems," *International Conference on Machine Learning*, pp. 2606–2615, 2018.

[50] A. R. Klivans and A. A. Sherstov, "Cryptographic hardness for learning intersections of halfspaces," *J. Comput. Syst. Sci.*, vol. 75, no. 1, pp. 2–12, 2009.

[51] W. Kong, J. G. Melo, and R. D. Monteiro, "Complexity of a quadratic penalty accelerated inexact proximal point method for solving linearly constrained nonconvex composite programs," *arXiv preprint arXiv:1802.03504*, 2018.

[52] Y. Kopsinis, K. Slavakis, and S. Theodoridis, "Online sparse system identification and signal reconstruction using projections onto weighted l1 balls," *IEEE Transactions on Signal Processing*, vol. 59, no. 3, pp. 936–952, 2011.

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, ser. NIPS'12, 2012, pp. 1097–1105.

[54] G. Lan, *Lectures on Optimization Methods for Machine Learning*. Springer-Nature, 2019, preprint.

[55] G. Lan and R. D. C. Monteiro, "Iteration-complexity of first-order penalty methods for convex programming," *Mathematical Programming*, vol. 138, pp. 115–139, 2013.

[56] G. Lan and R. D. C. Monteiro, "Iteration-complexity of first-order augmented lagrangian methods for convex programming," *Mathematical Programming*, vol. 155(1-2), 511–547, 2016.

[57] G. Lan, "An optimal method for stochastic composite optimization," *Math. Program.*, vol. 133, no. 1-2, pp. 365–397, 2012.

[58] G. Lan, S. Lee, and Y. Zhou, "Communication-efficient algorithms for decentralized and stochastic optimization," *Mathematical Programming*, 2018.

[59] G. Lan, Z. Li, and Y. Zhou, "A unified variance-reduced accelerated gradient method for convex optimization," in *NeurIPS 2019 : Thirty-third Conference on Neural Information Processing Systems*, 2019, pp. 10 462–10 472.

[60] G. Lan and Y. Yang, "Accelerated stochastic algorithms for nonconvex finite-sum and multi-block optimization," *arXiv preprint arXiv:1805.05411*, 2018.

[61] G. Lan and Y. Zhou, "An optimal randomized incremental gradient method," *Math. Program.*, vol. 171, no. 1-2, pp. 167–215, 2018.

[62] G. Lan and Z. Zhou, "Algorithms for stochastic optimization with expectation constraints," *arXiv preprint arXiv:1604.03887*, 2016.

[63] C. Lemaréchal, A. S. Nemirovski, and Y. E. Nesterov, "New variants of bundle methods," *Mathematical Programming*, vol. 69, pp. 111–148, 1995.

[64] Q. Lin, R. Ma, and Y. Xu, "Inexact proximal-point penalty methods for non-convex optimization with non-convex constraints," *arXiv preprint arXiv:1908.11518*, 2019.

[65] Q. Lin, R. Ma, and T. Yang, "Level-set methods for finite-sum constrained convex optimization," in *Proceedings of the 35th International Conference on Machine Learning*, vol. 80, 2018, pp. 3112–3121.

[66] Q. Lin, S. Nadarajah, and N. Soheili, "A level-set method for convex optimization with a feasible solution path," *SIAM Journal on Optimization*, vol. 28, no. 4, pp. 3290–3311, 2018.

[67] Q. Lin, S. Nadarajah, N. Soheili, and T. Yang, "A data efficient and feasible level set method for stochastic convex optimization with expectation constraints," *Available at SSRN 3433280*, 2019.

[68] R. Livni, S. Shalev-Shwartz, and O. Shamir, "On the computational efficiency of training neural networks," in *Advances in Neural Information Processing Systems 27*, 2014, pp. 855–863.

[69] M. Luby and N. Nisan, "A parallel approximation algorithm for positive linear programming," in *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing, May 16-18, 1993, San Diego, CA, USA*, 1993, pp. 448–457.

[70] R. Ma, Q. Lin, and T. Yang, "Proximally constrained methods for weakly convex optimization with weakly convex constraints," *arXiv preprint arXiv:1908.01871*, 2019.

[71] M. W. Mahoney, S. Rao, D. Wang, and P. Zhang, "Approximating the solution to mixed packing and covering lps in parallel o(epsilonˆ{-3}) time," in *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, 2016, 52:1–52:14.

[72] O. Mangasarian and S Fromovitz, "The fritz john necessary optimality conditions in the presence of equality and inequality constraints," *Journal of Mathematical Analysis and Applications*, vol. 17, pp. 37–47, 1967.

[73] P. Manurangsi and D. Reichman, "The computational complexity of training relu(s)," *CoRR*, vol. abs/1810.04207, 2018. arXiv: 1810.04207.

[74] J. M. Martínez and B. F. Svaiter, "A practical optimality condition without constraint qualifications for nonlinear programming," *Journal of Optimization Theory and Applications*, vol. 118, no. 1, pp. 117–133, 2003.

[75] N. Megiddo, "On the complexity of polyhedral separability.," *Discrete and Computational Geometry*, vol. 3, no. 4, pp. 325–338, 1988.

[76] A. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *Trans. Audio, Speech and Lang. Proc.*, pp. 14–22, 2012.

[77] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1574–1609, 2009.

[78] A. Nemirovski, "Prox-method with rate of convergence o(1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems," *SIAM Journal on Optimization*, vol. 15, no. 1, pp. 229–251, 2004.

[79] A. Nemirovsky and D. B. Yudin, "Problem complexity and method efficiency in optimization (a. s. nemirovsky and d. b. yudin)," *SIAM Review*, vol. 27, no. 2, pp. 264–265, 1985.

[80] Y. Nesterov, "Smooth minimization of non-smooth functions," *Math. Program.*, vol. 103, no. 1, pp. 127–152, 2005.

[81] Y. Nesterov, "Dual extrapolation and its applications to solving variational inequalities and related problems," *Math. Program.*, vol. 109, no. 2-3, pp. 319–344, 2007.

[82] Y. Nesterov, "Efficiency of coordinate descent methods on huge-scale optimization problems," *SIAM Journal on Optimization*, vol. 22, no. 2, pp. 341–362, 2012.

[83] Y. Nesterov, "Gradient methods for minimizing composite functions," *Mathematical Programming*, vol. 140, no. 1, pp. 125–161, 2013.

[84] Y. Nesterov, *Lectures on convex optimization*. Springer, 2018.

[85] M. Nouiehed, M. Sanjabi, J. D. Lee, and M. Razaviyayn, "Solving a class of non-convex min-max games using iterative first order methods," *arXiv preprint arXiv:1902.08297*, 2019.

[86] Y. Ouyang and Y. Xu, *Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems*, 2018. arXiv: `1808.02901 [math.OC]`.

[87] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[88] N. H Pham, L. M Nguyen, D. T Phan, and Q. Tran-Dinh, "Proxsarah: An efficient algorithmic framework for stochastic composite nonconvex optimization," *arXiv preprint arXiv:1902.05679*, 2019.

[89] S. A. Plotkin, D. B. Shmoys, and É. Tardos, "Fast approximation algorithms for fractional packing and covering problems," *Math. Oper. Res.*, vol. 20, no. 2, pp. 257–301, 1995.

[90] B. Polyak, "A general method of solving extremum problems," *Soviet Mathematics Doklady*, vol. 8(3), 593–597, 1967.

[91] H. Rafique, M. Liu, Q. Lin, and T. Yang, "Non-convex min-max optimization: Provable algorithms and applications in machine learning.," *arXiv preprint arXiv:1810.02060*, 2018.

[92] B. D. Rao and K. Kreutz-Delgado, "An affine scaling methodology for best basis selection," *IEEE Transactions on Signal Processing*, vol. 47, no. 1, pp. 187–200, Jan. 1999.

[93] S. J. Reddi, A. Hefny, S. Sra, B. Póczós, and A. J. Smola, "Stochastic variance reduction for nonconvex optimization," *International Conference on Machine Learning*, pp. 314–323, 2016.

[94] H. Robbins and D. Siegmund, "A convergence theorem for non negative almost supermartingales and some applications," *Optimizing Methods in Statistics*, pp. 111–135, 1971.

[95] G. Scutari, F. Facchinei, L. Lampariello, S. Sardellitti, and P. Song, "Parallel and distributed methods for constrained nonconvex optimization-part ii: Applications in communications and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1945–1960, 2017.

[96] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[97] O. Shamir, "Distribution-specific hardness of learning neural networks," *CoRR*, 2016.

[98] X. Shen, S. Diamond, Y. Gu, and S. Boyd, "Disciplined convex-concave programming," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, 2016, pp. 1009–1014.

[99] X. Shen, S. Diamond, Y. Gu, and S. Boyd, "Disciplined convex-concave programming," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, IEEE, 2016, pp. 1009–1014.

[100] J. Sherman, "Area-convexity, l_∞ regularization, and undirected multicommodity flow," in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, 2017, pp. 452–460.

[101] L. Song, S. Vempala, J. Wilmes, and B. Xie, "On the complexity of learning neural networks," *CoRR*, 2017.

[102] Y. Sun, P. S. Babu, and D. P. Palomar, "Majorization-minimization algorithms in signal processing, communications, and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 794–816, 2017.

[103] H. L. Thi, T. P. Dinh, H. Le, and X. Vo, "Dc approximation approaches for sparse optimization," *European Journal of Operational Research*, vol. 244, no. 1, pp. 26–46, 2015.

[104] H. A. L. Thi and T. P. Dinh, "DC programming and DCA: Thirty years of developments," *Mathematical Programming*, vol. 169, no. 1, pp. 5–68, 2018.

[105] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[106] M. J. Wainwright, "Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (lasso)," *IEEE transactions on information theory*, vol. 55, no. 5, pp. 2183–2202, 2009.

[107] D. Wang, S. Rao, and M. W. Mahoney, "Unified acceleration method for packing and covering problems via diameter reduction," in *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, 2016, 50:1–50:13.

[108] X. Wang, S. Ma, and Y. Yuan, "Penalty methods with stochastic approximation for stochastic nonlinear programming," *Mathematics of Computation*, vol. 86 (306), pp. 1793–1820, 2017.

[109] Z. Wang, K. Ji, Y. Zhou, Y. Liang, and V. Tarokh, "Spiderboost: A class of faster variance-reduced algorithms for nonconvex optimization," *arXiv preprint arXiv:1810.10690*, 2018.

[110] J. Weston, A. Elisseeff, B. Schölkopf, and M. Tipping, "Use of the zero-norm with linear models and kernel methods," *The Journal of Machine Learning Research*, vol. 3, pp. 1439–1461, 2003.

[111] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, 2009.

[112] L. Xiao and T. Zhang, "A proximal stochastic gradient method with progressive variance reduction," *SIAM Journal on Optimization*, vol. 24, no. 4, pp. 2057–2075, 2014.

[113] Y. Xu, "Iteration complexity of inexact augmented lagrangian methods for constrained convex programming," *Mathematical Programming*, 2019.

[114] Y. Xu, "Iteration complexity of inexact augmented lagrangian methods for constrained convex programming," *arXiv preprint arXiv:1711.05812*, 2017.

[115] N. E. Young, "Sequential and parallel algorithms for mixed packing and covering," in *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, 2001, pp. 538–546.

[116] N. E. Young, "Nearly linear-time approximation schemes for mixed packing/covering and facility-location linear programs," *CoRR*, vol. abs/1407.3015, 2014. arXiv: 1407.3015.

[117] H. Yu, M. Neely, and X. Wei, "Online convex optimization with stochastic constraints," *Advances in Neural Information Processing Systems*, pp. 1428–1438, 2017.

[118] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," *CoRR*, 2016.

[119] C.-H. Zhang, "Nearly unbiased variable selection under minimax concave penalty," *Annals of Statistics*, vol. 38, no. 2, pp. 894–942, 2010.

[120] C.-H. Zhang, J. Huang, *et al.*, "The sparsity and bias of the lasso selection in high-dimensional linear regression," *The Annals of Statistics*, vol. 36, no. 4, pp. 1567–1594, 2008.

[121] C.-H. Zhang and T. Zhang, "A general theory of concave regularization for high-dimensional sparse estimation problems," *Statistical Science*, vol. 27, no. 4, pp. 576–593, 2012.

[122] P. Zhao and B. Yu, "On model selection consistency of lasso," *Journal of Machine learning research*, vol. 7, no. Nov, pp. 2541–2563, 2006.

[123] D. Zhou, P. Xu, and Q. Gu, "Stochastic nested variance reduction for nonconvex optimization," in *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*, ser. NIPS'18, Montréal, Canada: Curran Associates Inc., 2018, pp. 3925–3936.

[124] E. Zurel and N. Nisan, "An efficient approximate allocation algorithm for combinatorial auctions," in *Proceedings 3rd ACM Conference on Electronic Commerce (EC-2001), Tampa, Florida, USA, October 14-17, 2001*, 2001, pp. 125–136.

VITA

Digvijay Boob was born on July 10, 1993 in Jalgaon, Maharashtra, India. He obtained his B.S. in Mechanical Engineering from Indian Institute of Technology, Bombay in 2014. He then worked as a high frequency trader for a year in Singapore. In August 2015, he enrolled at Georgia Institute of Technology. He completed his Ph.D. degree in the interdisciplinary Algorithms, Combinatorics, and Optimization program with home unit in Industrial and Systems Engineering department in July 2020. He became an assistant professor in the Engineering Management, Information, and Systems department at Southern Methodist University since August 2020.